

VŠB - TECHNICKÁ UNIVERZITA OSTRAVA  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY  
KATEDRA INFORMATIKY

**Zpracování dat pro zabezpečení videosekvencí pomocí  
MOTION JPEG 2000**

Data Processing of Digital Videosequence  
by MOTION JPEG 2000



## **PROHLÁŠENÍ**

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne .....

.....

Kamil Perutka

## **PODĚKOVÁNÍ**

Tímto bych rád poděkoval mému vedoucímu diplomové práce prof. Ing. Karlu Vlčkovi, Csc. a také Ing. Františkovi Kyselému za cenná doporučení, rady a pomoc, které mi při vypracování práce ochotně poskytli.

## ABSTRAKT

Tato práce obsahuje postup zpracování dat, jež je třeba pro aplikaci zabezpečovacích mechanismů, které je možno použít pro JPEG 2000, respektive Motion JPEG 2000. Tento princip kódování videosekvence užívá intra-frame kódování, což znamená, že sekvence snímků je sled samostatně kódovaných obrazů ve formátu JPEG 2000. Ten poskytuje nejen lepší kompresní poměr, ale hlavně nové požadované vlastnosti, které umožňuje klíčová část kompresního algoritmu Diskrétní vlnková transformace. Samotné zabezpečovací metody jsou rozděleny na dva typy: metody již implementované do základních částí standardu a metody maskovací, které mají za úkol minimalizovat projevy vzniklých chyb. V textu je popis implementovaných technik jako JPSEC nebo JPWL z části 8, respektive části 11 standardu. Do skupiny maskovacích metod patří metoda LL subband, filtrační techniky a detekce hran. Součástí práce je i praktické řešení uvedených metod pomocí dostupných knihoven, výběr z těchto metod a jejich aplikace na vybrané videosekvence. Díky měření kvality komprese pomocí PSNR jsou pro výsledky testování uvedeny možnosti kombinace daných metod a jejich největší účinnost včetně problematiky výpočetní náročnosti aplikovaných metod.

## KLÍČOVÁ SLOVA

*JPEG 2000, Motion JPEG 2000, Zabezpečení, Maskovací metody, Diskrétní vlnková transformace, LL subband, Filtrace obrazu, Hranové detektory*

## **ABSTRACT**

The thesis contains procedure of data processing, that is needed for application of security mechanisms, that are possible to use for JPEG 2000 or Motion JPEG 2000. This principle of videosequence coding uses an intra-frame coding, which means that sequence of frames is a sequence of separately coded pictures in JPEG 2000 format. This format provides better compression rate and new required properties, which are enabled by main part of compression algorithm Discrete wavelet transformation. Main security methods are divided into two types: the methods already implemented into basic parts of a standard and the concealment methods, that minimize all of errors. There is a description of implemented techniques like JPSEC in part 8 or JPWL in part 11 of the standard, in the text. The LL subband method, the filtering techniques and the edge detections belong to the concealment group methods. The thesis contains a practical solution of mentioned methods by help of accessible libraries, selection of these methods and their application on chosen videosequences. Because of quality compression measuring by PSNR, possible combinations of methods and their highest efficiency, including problems with computing complexity, are mentioned for the testing results.

## **KEYWORDS**

*JPEG 2000, Motion JPEG 2000, Security, Error Concealment Methods, Discrete wavelet transform, LL subband, Image filtering, Edge detectors*

## SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

CDF	Cohen-Daubechies-Feauveau
DCT	Discrete Cosine Transform - Diskrétní kosinova transformace
DWT	Discrete Wavelet Transform - Diskrétní vlnková transformace
DZQ	Deathzone Scalar Quantization – Skalární kvantování s mrtvou zónou
EBCOT	Embedded Block Coding with Optimal Truncation
ECM	Error Concealment Methods – Maskovací metody
EPB	Error Protecting Block – Blok chybové ochrany JPWL
EPC	Error Protecting Capability – Schopnost chybové ochrany JPWL
EPH	End of Packet – Konec paketu
ESD	Error Sensitivity Descriptor – Popisovač chybové citlivosti JPWL
GUI	Graphical User Interface – Grafické uživatelské prostředí
HD	High Definition - Vysoké rozlišení (1920px x 1080px)
ICT	Irreversible Color Transformation – Nevratná barevná transformace
IDWT	Inverse Discrete Wavelet Transform – Inverzní diskrétní vlnková transformace
IEC	International Electrotechnical Commission – Mezinárodní elektrotechnická komise
INSEC	Segmentová značka JPSEC syntaxe
ISO	International Organization for Standardization – Mezinárodní standardizační organizace
ITU	International Telecommunication Union – Mezinárodní telekomunikační unie
JP3D	JPEG 2000 Volumetric Imaging – Prostorové obrazy JPEG 2000
JPEG	Joint Photographic Experts Group
JPIP	JPEG 2000 Interactive protokol – Klient-Server protokol
JPSEC	JPEG 2000 Security – Zabezpečení JPEG 2000

JPWL	JPEG 2000 Wirelesss
LPCM	Linear Pulse Code Modulation – Lineární pulsní kódová modulace
LSI	Linear Space Invariant Systém – Prostorově invariantní lineární systém
MPEG	Motion Picture Experts Group
MSE	Mean Square Error – Střední kvadratická chyba
PDA	Personal Digital Assistant – Kapesní počítač
PSNR	Peak-Signal to Noise Ration – Špičkový poměr signálu k šumu
RA	Registration Authority JPSEC – Registrační autorita pro JPSEC
RCT	Reversable Color Transformation – Vratná barevná transformace
RED	Residual Error Descriptor – Popisovač zbytkových chyb JPWL
RGB	Označení barevného prostoru – Red Green Blue (červená zelená modrá)
ROI	Region Of Interest – Oblast zájmu
SEC	Segmentová značka JPSEC syntaxe
SNR	Signal to Noise Ration – Poměr signálu k šumu
SOP	Start of Packet – Začátek paketu
WMV	Windows Media Video



# OBSAH

<b>1</b>	<b>ÚVOD.....</b>	<b>1</b>
<b>2</b>	<b>STANDARD JPEG 2000 .....</b>	<b>3</b>
2.1	JPEG 2000.....	3
2.2	Vlastnosti standardu JPEG 2000.....	5
2.3	Srovnání JPEG 2000 se starším JPEG .....	6
2.4	Postup komprese standardu JPEG 2000.....	6
2.4.1	Předzpracování dat.....	7
2.4.2	Rozklad obrazu .....	8
2.4.3	Barevná transformace .....	8
2.4.4	Diskrétní vlnková transformace DWT.....	10
2.4.5	Princip diskrétní vlnkové transformace .....	10
2.4.6	Diskrétní vlnková transformace DWT2.....	14
2.4.7	Kvantování.....	16
2.4.8	Entropické kódování a tvorba souboru dat .....	17
<b>3</b>	<b>ZABEZPEČOVACÍ METODY PRO MOTION JPEG 2000.....</b>	<b>19</b>
3.1	Motion JPEG 2000.....	19
3.2	Zabezpečení formátu proti přenosovým chybám .....	20
3.3	Posuzování vizuální kvality komprese .....	21
3.4	Implementované zabezpečovací mechanismy .....	22
3.4.1	Part 1 – Core Coding Systém.....	22
3.4.2	Part 8 – JPSEC .....	23
3.4.3	Part 11 – JPWL .....	24
3.5	Metoda LL Subband.....	26
3.6	Zajištění datového toku Motion JPEG 2000 .....	28
3.7	Filtrační metody .....	28
3.7.1	Lineární filtry .....	28
3.7.2	Mean Filtr – Průměrovací Filtr .....	29
3.7.3	Kruhový filtr .....	30

3.7.4	Rekurzivní filtry.....	31
3.7.5	Filtrační metody – nelineární.....	31
3.7.6	Mediánový filtr .....	31
3.8	Detekce Hran.....	33
3.8.1	Gradientní operátory .....	33
3.8.2	Hranové detektory.....	35
3.8.3	Robertsův operátor.....	36
3.8.4	Operátor Prewittové .....	36
3.8.5	Sobelův operátor .....	37
3.8.6	Robinsonův operátor .....	38
3.8.7	Kirschův operátor .....	38
3.8.8	Laplaceův operátor .....	38
3.8.9	Logaritmický operátor .....	40
3.8.10	Cannyho hranový detektor.....	41
<b>4</b>	<b>NÁVRH ŘEŠENÍ.....</b>	<b>44</b>
4.1	Zvolené zabezpečovací metody .....	45
<b>5</b>	<b>TESTOVÁNÍ NA ZVOLENÝCH VIDEOSEKVENCÍCH .....</b>	<b>46</b>
5.1	Metoda LL Subband.....	46
5.2	Metoda filtrační .....	49
5.3	Metoda Nahrazování .....	51
5.4	Metody hranových detektorů .....	52
5.5	Další výsledky.....	53
<b>6</b>	<b>NÁROČNOST PRO MOBILNÍ TECHNOLOGIE.....</b>	<b>54</b>
<b>7</b>	<b>ZÁVĚR .....</b>	<b>56</b>
	<b>SEZNAM POUŽITÉ LITERATURY.....</b>	<b>58</b>
	<b>SEZNAM OBRÁZKŮ .....</b>	<b>60</b>
	<b>SEZNAM PŘÍLOH.....</b>	<b>61</b>
	<b>PŘÍLOHA Č. 1.....</b>	<b>62</b>
	<b>PŘÍLOHA Č. 2.....</b>	<b>63</b>
	<b>PŘÍLOHA Č. 3.....</b>	<b>64</b>

# 1 ÚVOD

Obrazový vjem je pro člověka důležitý a dokáže jej vnímat velice přesně. Snaha přenést velké množství informací ze všech oblastí světa právě do obrazu je a bude velká. Už před půl stoletím se při vývoji informačních kanálů předpokládalo přenášení obrazu a nyní obrazové informace zabírají velkou část objemu přenášeného ve všech typech sítí. Stejně tak rostou obrovskou rychlostí požadavky na multimediální technologie, neustále se vyvíjí nové technologie a nové kompresní algoritmy. Při zvyšujícím se množství multimediálních dat vznikají větší nároky na jejich vlastnosti, jako je zabezpečení, malý datový tok při současných vysokých HD rozlišeních, odolnost proti chybovosti, volba mezi ztrátovou a bezztrátovou kompresí.

JPEG 2000 je kompresní standard vyvinutý Joint Photographic Experts Group (JPEG). Předchozí JPEG standard se stal celosvětově užívanou technologií, ať už v oblasti digitálních fotografií nebo internetu. JPEG 2000 jako nástupce tohoto zaběhnutého standardu má vylepšené kompresní algoritmy a nabízí nové požadované vlastnosti. Klíčovým kompresním prvkem v algoritmu JPEG 2000 je diskretní vlnková transformace (DWT). Právě z tohoto článku kompresního řetězce vychází většina odlišných vlastností standardu. Z obrazového formátu JPEG 2000 přímo vychází videoformát Motion JPEG 2000. Ten je prakticky tvořen sledem jednotlivých obrazů kódovaných v základním JPEG 2000 formátu. Tento kontinuální tok obrazů tvoří výslednou videosekvenci.

Přes všechny výše uvedené předpoklady existují i v současnosti datové sítě, u kterých není zabezpečení při přenosu dat věnovaná dostatečná pozornost ať už z důvodu nákladnosti vývoje takových sítí nebo z důvodu nedostatečné šířky přenosového pásma. Právě v těchto sítích vznikají při přenosu dat chyby různého charakteru.

Součástí všech navrhovaných standardů jsou tedy dané zabezpečovací mechanismy, které jsou obsaženy již ve specifikacích daného kompresního algoritmu. Motion JPEG 2000 není výjimkou a chybová odolnost tohoto formátu včetně použití zavedených metod je dále v textu popsána. V tomto případě se tedy jedná o zabudované možnosti zabezpečení dat, snažící se chybám předcházet již při tvorbě datového toku.

Dalšími metodami jsou ty, které počítají s nedokonalostí implementovaných technik a přenosových kanálů a jejich úkolem je následující pokus o rekonstrukci. Techniky tohoto druhu jsou maskovací metody, které chyby při přenosu neodstraňují, ale snaží se eliminovat jejich důsledek pro vnímání člověka. Pro toto odstranění vzniklých chyb existují dva způsoby. Prvním je využít značnou

podobnost po sobě jdoucích snímků. Z této podobnosti vychází možnost extrapolace obrazových dat z jednoho obrazu a jejich následné využití k maskování chyb v poškozeném obrazu. Druhá maskovací metoda používá k eliminaci chyb obrazové informace v samotném poškozeném snímku a využívá k tomu lineárních a nelineárních obrazových filtrů. S použitím filtračních metod souvisí problém detekce hran. Popsány jsou jednotlivé hranové detektory. Dále je nastíněna možnost nahrazování snímků k zachování celistvosti videosekvence. Jsou popsány principy pro hodnocení kvality. Uvedeny jsou metody objektivní i subjektivní.

Pro pochopení funkce zabezpečení obrazu proti chybám nebo jejich následné eliminace musíme nejprve důkladně porozumět postupu kompletního řetězce kompresního algoritmu JPEG 2000, na němž je videoformát Motion JPEG 2000 založen. Bude uvedeno všech jednotlivých 12 částí standardu a přednosti formátu oproti svému předchůdci JPEG. Dále následuje postup komprese JPEG 2000 vždy se vztahem na videoformát Motion JPEG 2000. Jednotlivé kapitoly jsou věnovány všem článkům komprimačního řetězce, přičemž větší část bude věnována transformaci DWT. Po tomto základním úvodu následuje seznámení se s konkrétními zabezpečovacími metodami pro Motion JPEG 2000. Nejprve jsou uvedeny principy implementované přímo do jednotlivých částí standardu JPEG 2000. Následují metody maskovací, tedy LL subband, metody filtrace obrazu, metody detekce a obnovení hran.

Dle dostupných knihoven je dále zvoleno řešení k aplikaci těchto metod na reálné videosekvence. Při návrhu byly vybírány maskovací metody uvedené dříve v textu, s možností ovlivnit jejich parametry pro správnou interpretaci jejich funkce. Poté jsou zmíněné videosekvence otestovány s možností porovnávat mezi sebou jednotlivé výsledky všech metod. Je zvoleno několik druhů videosekvencí dle jejich dynamičnosti. I přes snahu vybírat pro názornost vhodná videa, je tento typ multimediálního obsahu vhodnější hodnotit v elektronické podobě na přiloženém CD. Následuje vyjádření k problému výpočetní náročnosti aplikovaných metod se vztahem k mobilním technologiím s uvedením méně náročné implementace klíčové části kompresního řetězce, tedy DWT.

## 2 STANDARD JPEG 2000

### 2.1 JPEG 2000

Formát JPEG 2000 byl vytvořen skupinou JPEG, která představuje uskupení akademických institucí a firem. Byla vytvořena v roce 1986 organizacemi ISO a ITU a měla za úkol vytvořit standard pro obrazová data. Jejich výsledkem je mezinárodní standard ISO/IEC 10918-1 známý jako JPEG. Ten využíval diskrétní kosinovou transformaci, která redukuje zobrazovací body do čtvercových bloků o hraně 8 pixelů. Ovšem tím, jak v devadesátých letech internet a multimediální obsah zasahoval čím dál více do všech sfér života, vznikly nové požadavky na takový standard a JPEG se stal pro některé oblasti použití nedostačujícím [1].

V roce 1997 se tedy začal vyvíjet nový kompresní standard založený na diskrétní vlnkové transformaci. Jeho vytvoření přineslo některé nové vylepšení do sféry obrazových formátů. Při jeho tvorbě však nebylo cílem vytvořit nejefektivnější kompresní algoritmus, avšak vytvořit formát, který bude univerzální pro všestranné použití, bude podporovat ztrátovou i bezztrátovou kompresi. Finální verze JPEG 2000 je z roku 2000. Nový způsob komprese odstraňuje právě nepříjemné bloky 8x8 známé z DCT komprese a dosahuje větší úrovně komprese při současné menší ztrátě kvality. Další velká výhoda J2K tkví v tom, že lze kompresní poměr nastavit při otevření [2].

Specifikace JPEG 2000 je rozdělena do 12 částí:

- **Part 1: Core Coding System**

přesně definuje základní kódování a poskytuje informace o většině standardních možností, poskytuje základní informace pro vytvoření funkční implementace

- **Part 2: Extensions**

zahrnuje zlepšení kompresního algoritmu použitím jiného kodeku a další pokročilé vlastnosti formátu

- **Part 3: Motion JPEG 2000**

část popisující videoformát Motion JPEG 2000

- **Part 4: Conformance**  
definuje normy pro testování implementací a jejich shodu se standardem ISO/IEC 15444-1
- **Part 5: Reference software**  
referenční implementace v jazyce Java a v jazyce C
- **Part 6: Compound image file format**  
specifikuje souborový formát
- **Part 7**  
byla dle specifikace vypuštěna
- **Part 8: JPSEC – Security**  
obsahuje specifikaci JPSEC – JPEG 2000 Security
- **Part 9: JPIP – Interactive protocols**  
specifikuje Client-Server protokol s názvem JPIP – JPEG 2000 Interactive Protocol
- **Part 10: JP3D – Volumetric Imaging**  
dle specifikace stále ve fázi pracovního návrhu, týká se rozšíření z rovinných na prostorové obrazy, kódování třídímenzionálních dat
- **Part 11: JPWL – Wireless Applications**  
určena pro bezdrátové aplikace, zahrnuje pokročilé metody zabezpečení proti poruchám při přenosu
- **Part 12: ISO Base Media File Format**  
spolupráce mezi JPEG a MPEG k vytvoření souborového formátu pro budoucí aplikace [3]

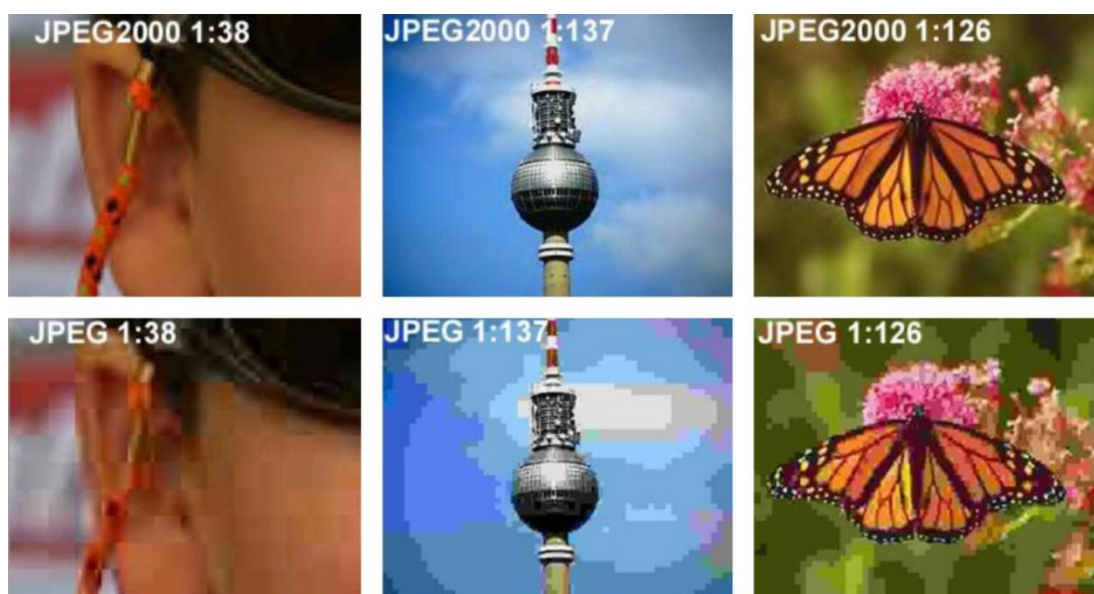
## 2.2 Vlastnosti standardu JPEG 2000

Výhody formátu JPEG 2000 v podstatě vychází z výše uvedeného rozdílu oproti JPEG, a to v použití DWT:

- **Účinnost kompresního algoritmu** – vyšší účinnost komprese oproti standardu JPEG při současném zachování stejné kvality obrazu, lepší komprese převážně v nízkých bitových hloubkách
- **Ztrátová a bezztrátová komprese** – ve vědních oblastech nebo například ve filmové produkci byl často kladen nárok právě na bezztrátovou kompresi
- **Podpora více rozlišení** – v jednom komprimovaném souboru lze pomocí použití DWT uchovat několik rozlišení původního obrazu
- **Odolnost vůči chybám** – díky postupům uvedeným dále je JPEG 2000 odolnější například proti zarušenému prostředí přenosu
- **Zpracování dat ve zkomprimované podobě** – formát umožňuje náhodný přístup k datům, dále lze ořezávat nebo otáčet obraz bez dekompresního procesu
- **Schopnost komprese** – komprese nejen přirozených obrazů, ale i počítačové grafiky, obrazů z vědních oblastí – medicína, atd.
- **Posunutí a přiblížení v obraze** – operace pan a zoom lze provádět s dekompresí jen k tomu potřebných částí
- **Kódování oblasti zájmu** – je-li v závislosti na obrazu zapotřebí zobrazit některé oblasti přesněji než zbytek obrazu, lze využít tzv. ROI (Region Of Interest), což umožní tyto oblasti kódovat s vyšší kvalitou než ostatní
- **Progresivní přenášení dat** – umožňuje postupnou rekonstrukci obrazu, přenos se zvyšující se kvalitou, obraz je nejprve rychle zobrazen v základním rozlišení a postupně se jeho kvalita s přicházejícími daty zvyšuje a zvětšuje se jeho rozlišení
- **Algoritmus vhodný i pro velké kompresní poměry**
- **Ochrana autorských práv** – je realizováno technikou vodoznaků a šifrování

## 2.3 Srovnání JPEG 2000 se starším JPEG

Při běžně používaném středním stupni komprese nabízí JPEG 2000 cca o 20% až 30% lepší výsledky komprese ve srovnání se standardem JPEG. Požadujeme-li stupeň komprese nízký nebo naopak vysoký, bude procentuální srovnání ještě výhodnější ve prospěch novějšího JPEG 2000. Při použití bezztrátové varianty je dosaženo kompresního poměru 1:2 [4]. U obrazů s vyšším kompresním poměrem se projevuje vyšší rozmazanost než u původního obrazu. Nejlepších výsledků se dosahuje u snímků s nevýraznými a nekонтastními okraji (hranami v obraze), toho se využívá například v oblasti lékařství pro lékařské snímky. Neprojevují se artefakty vzniklé u JPEG 2000 dělením bloků 8x8 px. Často se využívá metoda chroma subsampling, při které se barevným komponentům přiřadí nižší rozlišení a využije se nedokonalosti lidského oka. Než na barvu má totiž vyšší citlivost na jas [4]. Ve srovnání s JPEG má nový standard vyšší časovou a paměťovou náročnost.



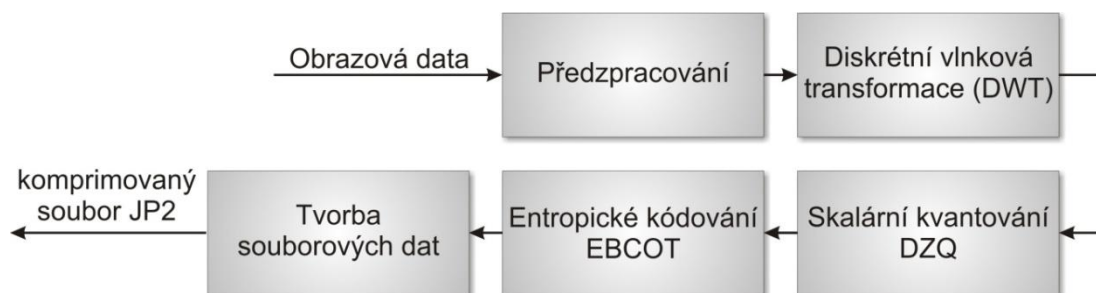
Obr. 1: Ilustrativní srovnání kompresí JPEG a JPEG 2000

## 2.4 Postup komprese standardu JPEG 2000

První část standardu specifikuje postup komprimace obrazových dat, jak jsem zmiňoval výše. Prvním článkem řetězce komprese je předzpracování dat, kde je proveden případný rozklad obrazu a barevná transformace. Následuje stěžejní část algoritmu, kterou je DWT, na jejímž výstupu jsou



koeficienty jednotlivých subpásem. Ty jsou v dalším funkčním bloku kvantování pomocí skalárního kvantování DZQ. Kvantované koeficienty jsou dále přijaty blokovým kóděm a následuje část pro formování výsledného datového toku, po které jsou data připravena na výstupu komprese [5]. Jednotlivé bloky kompresního celku jsou uvedeny na následujícím obrázku 2.



Obr. 2: Blokové schéma kompresního algoritmu JPEG 2000 [6]

### 2.4.1 Předzpracování dat

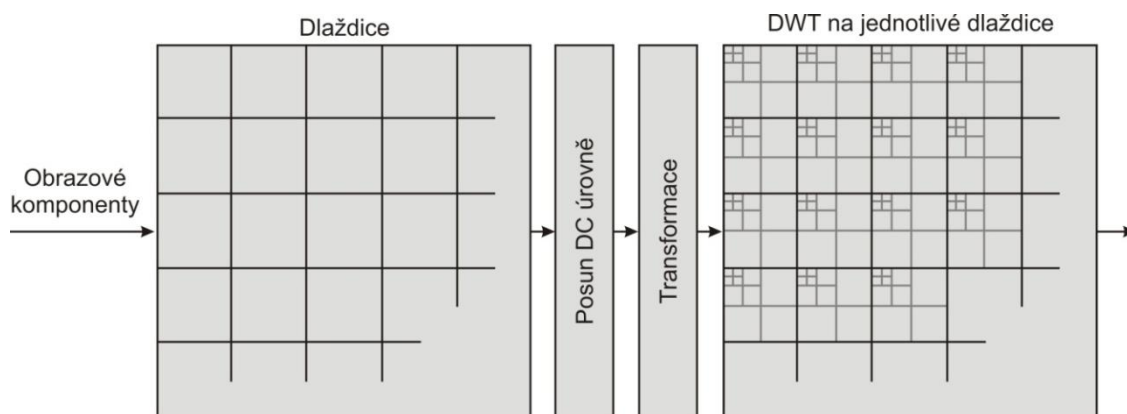
Před hlavním blokem, ve kterém je provedena DWT, musí být data převedená do vhodného formátu. Vstupní obrazová data lze reprezentovat jako dvourozměrné pole hodnot:

$$x[n_1, n_2], \quad 0 \leq n_1 \leq N_1 \quad 0 \leq n_2 \leq N_2$$

kde  $N_1$  a  $N_2$  jsou hodnoty ve vertikálním a horizontálním směru.  $n_1$  určuje řádek a  $n_2$  nám udává pořadí pixelu na tomto řádku. Za počátek tohoto označení je brán levý horní roh obrazu. Transformační výpočet probíhá dle toho, zda byla zvolena ztrátová nebo bezztrátová komprese. Ztrátová transformace probíhá nad reálnými koeficienty, jejichž číselná přesnost je na závěr zmenšená použitím skalárního kvantování s mrtvou zónou DZQ [7]. Oproti tomu bezztrátová transformace pracuje pouze s celými čísly a využívá výhradně matematicky bezztrátové operace. Proto je u tohoto způsobu kódování na závěr vynecháno DZQ, které nevratně snižuje přesnost koeficientů [8]. Samotné předzpracování dat představuje rozklad obrazu, posun DC úrovně a vybraný typ barevné transformace.

### 2.4.2 Rozklad obrazu

Je-li obraz příliš velký na to, aby byl zpracováván vcelku nebo není-li výhodné takto obraz zpracovávat, je užito techniky rozdělení na menší samostatné celky. Ty jsou poté zpracovávány zcela samostatně. Tyto celky neboli dlaždice se samozřejmě nesmějí překrývat [5]. Rozklad na tyto celky je snadné řešení popsaného problému. Přináší výhody jako paralelitu algoritmu, ale s jistým rizikem. Čím menší je totiž plocha těchto celků, tím klesá kompresní výkon po stránce odstupů signálu od šumu. To lze potom pozorovat i v kvalitě výsledného obrazu. Tuhle metodu lze přirovnat ke zpracování obrazu ve starším JPEGu, který používá diskretní kosinovou transformaci, která pracuje s bloky 8x8 pixelů. Poté dojde k posunu DC úrovně, při které se od všech vzorků obrazových složek odečítá konstanta  $2^{p-1}$ , kde  $p$  udává počet bitů obsažených v jednom vzorku obrazové složky. Na straně dekodéru se inverzně konstanta přičítá [6].



Obr. 3: Rozložení snímku na dlaždice [6]

### 2.4.3 Barevná transformace

Jako další krok následuje barevná transformace. Barevná transformace je aplikovatelná pouze na první tři barevné komponenty, které zároveň musí mít shodné rozměry a shodnou bitovou hloubku. Tyto komponenty jsou v podstatě složky červená, zelená a modrá z barevného prostoru RGB [8]. JPEG 2000 zpracovává jednotlivé komponenty odděleně a díky tomu mezi barvami nerozlišuje. Převodem na jiný barevný prostor lze v případě RGB snížit redundanci v barevné informaci a tím zmenšit objem dat.

## Nevratná barevná transformace (ICT)

Nevratná barevná transformace definovaná v 1. části standardu JPEG 2000 využívá běžnou konverzi z barevného prostoru RGB do YCbCr. Tu můžeme vyjádřit následovně:

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.16875 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & 0.081312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Jak je ze vztahů patrné, nejvíce informací je tedy obsaženo v jasové složce Y. Zbývající ,tzv. chromatické složky Cb a Cr, představují rozdíl mezi modrou složkou, respektive červenou, a jasnem [2]. To, že nejvyšší váhu obsahuje jasová složka Y, odpovídá vnímání lidského oka. Nejcitlivější je tento lidský orgán právě na zelenou barvu, jejíž informace nese jasová složka nejvíce. Je tomu tak proto, že lidské oko je mnohem citlivější na chyby v jasové složce v porovnání se složkami chromatickými.

Potřebnou zpětnou konverzi do barevného prostoru RGB definuje vztah:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.402 \\ 1 & -0.344136 & -0.714136 \\ 1 & 1.772 & 0 \end{bmatrix} \begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix}$$

## Vratná barevná transformace (RCT)

Reverzibilní transformace RCT je aplikována při použití bezztrátové komprese. Cílem tohoto kroku je deklarace barev a vymezení barevného prostoru pro efektivní kompresi a kvantizaci. Pro složky RGB představuje RCT aproximovanou YUV transformaci.

Převod z RGB prostoru je definován takto [6]:

$$\begin{bmatrix} Y_r \\ U_r \\ V_r \end{bmatrix} = \begin{bmatrix} \left( \frac{R + 2G + B}{4} \right) \\ R - G \\ B - G \end{bmatrix}$$

Zpětnou konverzi do RGB potom definuje vztah:

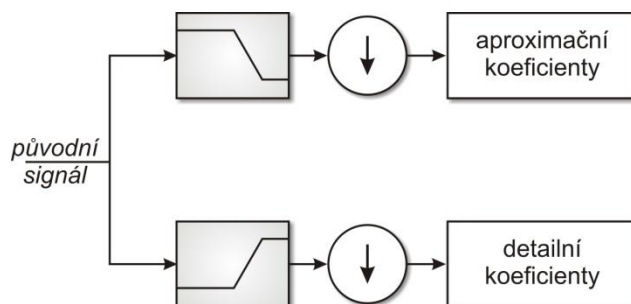
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} Y_r - \left( \frac{U_r + V_r}{4} \right) \\ V_r - G \\ U_r - G \end{bmatrix}$$

#### 2.4.4 Diskrétní vlnková transformace DWT

Tato metoda analýzy digitálního signálu má mnoho aplikací jak v oblasti informatiky, tak v dalších vědních oborech jako je strojírenství, matematika, atd. V oblasti informatiky je použita nejen pro kompresi, ale také například při zpracování řeči nebo videa. DWT poskytuje náhled na signál v rozličných rozlišeních jako na množiny hrubých i detailních hodnot [8]. DWT použita pro JPEG 2000 nahradila původní DCT použitou u JPEG. DCT by dle principu mohla být označena jako bloková, na rozdíl od toho DWT lze označit jako celoplošnou. Právě tento podstatný rozdíl poskytuje JPEG 2000 některé vlastnosti, kterými starší JPEG nedisponuje. Jedná se například o zakomponování více různých rozlišení do aktuálního datového toku. DWT lze na vstupní data aplikovat opakovaně. Počet takovýchto průchodů určuje jak kompresní poměr, tak kvalitu výsledného komprimovaného obrazu. Lze říci, že s vyšším množstvím průchodů snižujeme kompresní poměr, ale také zvyšujeme kvalitu obrazu [9]. Navíc právě celoplošné pojetí odstraňuje ony blokové artefakty známé u JPEG, zejména při vyšších kompresních poměrech. Dále pomocí celoplošné DWT je možné použít ztrátovou i bezztrátovou kompresi v jednom datovém toku. Princip diskrétní transformace je popsán v následující kapitole. Funkce je vysvětlena na jednorozměrné variantě DWT, po které následuje popis DWT2.

#### 2.4.5 Princip diskrétní vlnkové transformace

Chceme-li nejprve jednoduše vysvětlit průběh DWT, můžeme říci, že koeficienty vstupního signálu jsou přivedeny na vstupy dvojice filtrů. Dále jsou z obou takto filtrovaných posloupností odstraněny liché koeficienty – signál je tedy decimován. Tímto postupem vznikají nové koeficienty, které jsou kratší než vstupní signál. Filtry, použité v této fázi, jsou označovány jako dolní propust (low-pass filter) a horní propust (high-pass filter). Koeficienty vzniklé v části řetězce obsahující filtr horní propust jsou označovány jako detailní vlnkové koeficienty a koeficienty vzniklé z části filtru obsahující dolní propust jsou označovány aproximační vlnkové koeficienty. Postup DWT lze opakovat. Rozdílem od prvního průchodu je to, že vstupem do algoritmu nyní nebude vstupní vektor dat, ale vektor aproximačních vlnkových koeficientů. Počet těchto průchodů udává hloubku dekompozice [9]. Postup jednorozměrné dopředné DWT transformace je ilustrován na obrázku 4.



Obr. 4: Jednoduchá jednorozměrná 1D DWT analýza

Filtr, označovaný jako dolní propust, zachovává nižší frekvence signálu a zároveň eliminuje vysoké frekvence. Zcela analogicky filtr, označovaný jako horní propust, zachovává frekvence vysoké a eliminuje nízké. Jsou-li například tímto signálem koeficienty obrazu, pak po průchodu dolní propustí jsou zachovány koeficienty nízkých hodnot a opačně [8].

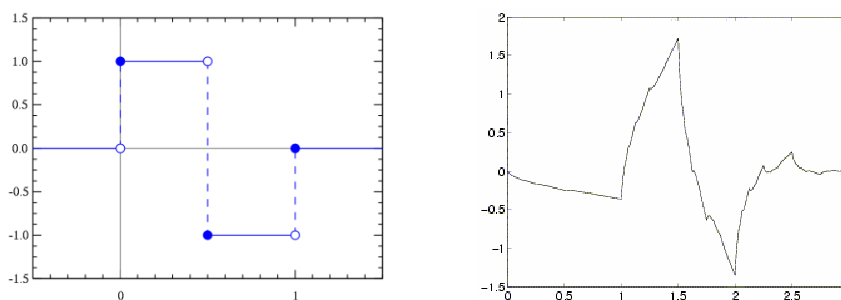
Pro názornost je uvedena jedna z dvojice filtrů užívaných k analýze signálu:

$$h_0(n) = (-1, 2, 6, 2, -1)/8$$

$$h_1(n) = (-1, 2, -1)/2$$

kde  $h_0(n)$  je filtr typu dolní propust a  $h_1(n)$  je filtr typu horní propust.

Dvojice filtrů, uvedená jako příklad, je označována jako biortogonální spline vlnky 5/3 nebo LeGall 5/3. Hodnota 5 značí filtr typu horní propust délky pět a parametr 3 pak znamená dolní propust délky tři [10]. Standard JPEG 2000 filtr LeGall používá pro bezztrátovou DWT transformaci.



Obr. 5: a) Haarova vlnka, b) Daubechies vlnka [11]

Když tedy aplikujeme dolní propust  $h_0(n)$  na signál  $x[n]$ , v  $i$ -tém kroku je stav následující:

$$x[i] = \frac{-1}{8}x[i-2] + \frac{2}{8}x[i-1] + \frac{6}{8}x[i] + \frac{2}{8}x[i+1] + \frac{-1}{8}x[i+2]$$

Výsledný počet hodnot vektoru vlnkových koeficientů je shodný s počtem hodnot původního signálu, což je způsobeno odstraněním lichých koeficientů z výstupu horní a dolní propusti.

Hodnoty filtru LeGall jsou tyto:

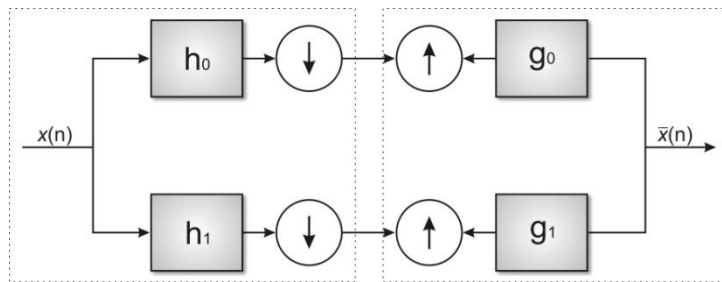
$$h_0(n) = \left( \frac{-1}{8} \quad \frac{1}{4} \quad \frac{3}{4} \quad \frac{1}{4} \quad \frac{-1}{8} \right)$$

$$h_1(n) = \left( \frac{-1}{2} \quad 1 \quad \frac{-1}{2} \right)$$

$$g_0(n) = \left( \frac{1}{2} \quad 1 \quad \frac{1}{2} \right)$$

$$g_1(n) = \left( \frac{-1}{8} \quad \frac{-1}{4} \quad \frac{3}{4} \quad \frac{-1}{4} \quad \frac{-1}{8} \right)$$

Obnovení hodnot originálního signálu pomocí získaných vlnkových koeficientů je zajištěno aplikací dvojice filtrů,  $g_0(n)$  a  $g_1(n)$  dle obrázku 6, které jsou určeny k syntéze. Před použitím filtrů je nutné doplnit všechny liché koeficienty (reverzní decimace), jež byly v předchozí fázi analýzy vypuštěny. Na jejich místo jsou doplněny nulové hodnoty. Poté je filtr horní propust použit na výstup filtru  $h_0(n)$  a analogicky filtr dolní propust je použit na výstup filtru  $h_1(n)$  [8]. Výsledný signál, který je shodný s původním vstupním signálem  $x(n)$ , získáme sloučením dvojice signálů vzniklých aplikací filtrů do jediného výstupního signálu  $\bar{x}(n)$ .



Obr. 6: Analýza a syntéza 1D DWT [12]

Filtr LeGall je ve standardu JPEG 2000 využit pro bezztrátovou variantu. Ztrátová využívá filtru s reálnými koeficienty biortogonální spline vlnky 9/7 neboli vlnky Cohen-Daubechies-Feauveau 9/7 (CDF 9/7), někdy zkráceně označován jako Daubechies filtr. Výhoda filtrů s reálnými

koeficienty spočívá v tom, že generují výstup, který lze úspěšněji zkomprimovat. Oproti tomu nevýhodou je ztrátovost celého procesu, což umožňuje pouze bezztrátová varianta.

CDF 9/7 analýza spočívá v aplikaci následujících filtrů:

$$h_0(n) = (h^{-4}h^{-3}h^{-2}h^{-1}h^0h^1h^2h^3h^4)$$

$$h_1(n) = (h^{-3}h^{-2}h^{-1}h^0h^1h^2h^3)$$

Konkrétní hodnoty koeficientů  $h_0(n)$  a  $h_1(n)$  jsou tyto [8]:

$h_0(n)$ :

$$h^0 = 0,602949018236$$

$$h^{-1} = h^1 = 0,266864118443$$

$$h^{-2} = h^2 = -0,078223266529$$

$$h^{-3} = h^3 = -0,016864118443$$

$$h^{-4} = h^4 = 0,026748757411$$

$h_1(n)$ :

$$h^0 = 1,11508705$$

$$h^{-1} = h^1 = -0,591271763114$$

$$h^{-2} = h^2 = -0,057543526229$$

$$h^{-3} = h^3 = 0,091271763114$$

Filtry užívané pro aplikaci CDF 9/7 syntézy jsou tyto:

$$g_0(n) = (g^{-3}g^{-2}g^{-1}g^0g^1g^2g^3)$$

$$g_1(n) = (g^{-4}g^{-3}g^{-2}g^{-1}g^0g^1g^2g^3g^4)$$

Pro úplnost jsou uvedeny také konkrétní hodnoty těchto filtrů [8]:

$g_0(n)$ :

$$g^0 = 1,11508705$$

$$g^{-1} = g^1 = 0,591271763114$$

$$g^{-2} = g^2 = -0,057543526229$$

$$g^{-3} = g^3 = -0,091271763114$$

$g_1(n)$ :

$$g^0 = 0,602949018236$$

$$g^{-1} = g^1 = -0,266864118443$$

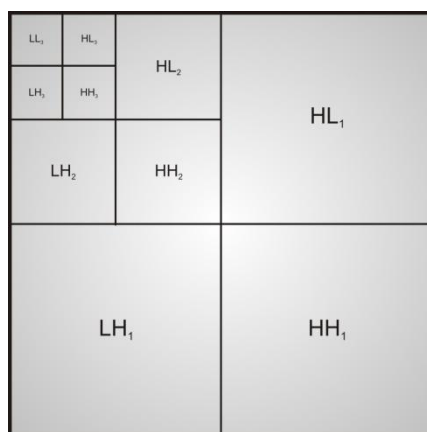
$$g^{-2} = g^2 = -0,078223266529$$

$$g^{-3} = g^3 = 0,016864118443$$

$$g^{-4} = g^4 = 0,026748757411$$

### 2.4.6 Diskrétní vlnková transformace DWT2

Pro použití tohoto způsobu transformace na dvourozměrná obrazová data v rámci komprese JPEG 2000, je třeba výše uvedený princip funkce DWT rozšířit o další rozměr na DWT2. Ta je určena pro analýzu a syntézu dvourozměrných dat. Toho se dosáhne použitím DWT transformace odděleně na řádky a sloupce. V prvním kroku se provede analýza pomocí filtru  $h_0$  a  $h_1$  na jednotlivé řádky obrazu separátně, včetně decimace. Stejné filtry jsou v dalším kroku použity znova, a to na analýzu výsledku předchozího kroku po sloupcích, ze kterých jsou na konci transformace opět odstraněny liché koeficienty. Z těchto kroků získáme 4 subpásma o stejné velikosti, respektive samostatné obrazy o celkově stejné velikosti jako vstupní obraz. Jak je vidět na následujícím obrázku, subpásma v levém horním rohu takto vzniklých subpásem lze dále stejným způsobem dekomponovat. Na obrázku 7 je uveden příklad tříúrovňové dekompozice a značení jednotlivých subpásem.



Obr. 7: Tříúrovňová DWT2 transformace

Tyto subpásma bývají také označovány jako  $cA, cH, cV, cD$  – koeficienty aproximativní, horizontálně detailní, vertikálně detailní a diagonálně detailní [13]. V našem případě použijeme označování  $LL, HL, LH, HH$ . Na uvedeném příkladu vznikly v první úrovni dekompozice 4 subpásma  $LL_1, HL_1, LH_1, HH_1$ . Při další úrovni bylo subpásma  $LL_1$  dále dekomponováno a vznikly subpásma  $LL_2, HL_2, LH_2, HH_2$ . Při tříúrovňové transformaci by se postupovalo dále stejným



způsobem, tentokrát by proběhla dekompozice subpásma  $LL_2$ . Výčet výsledných subpássem by tedy byl:

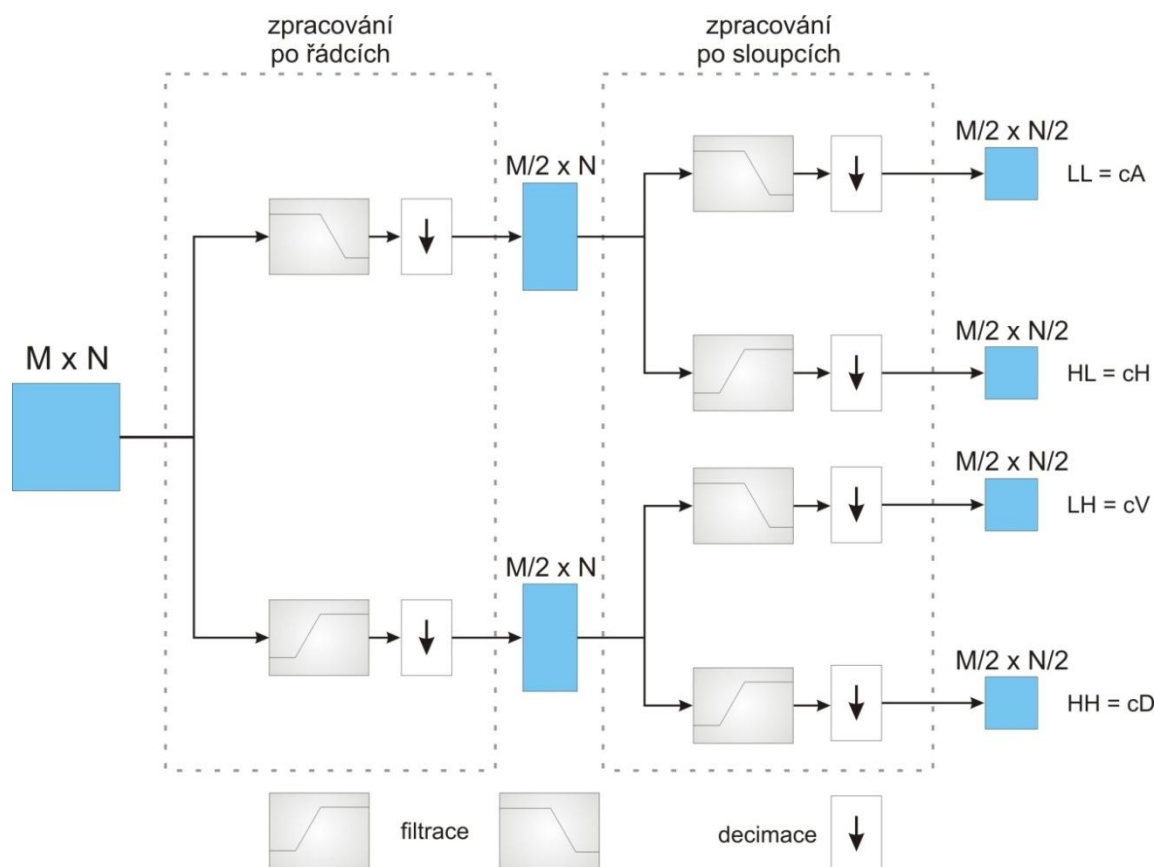
$$HL_1, LH_1, HH_1, HL_2, LH_2, HH_2, LL_3, HL_3, LH_3, HH_3$$



Obr. 8: Diskrétní vlnková transformace druhé úrovně (DWT2) [8]

V subpásmu LL je signál filtrován výhradně filtrem typu dolní propust a nachází se zde největší energie z originálního obrazu ve formě nízkofrekvenčních hodnot. Ty reprezentují přibližný tvar. Na zbývající subpásma LH, HL a LL byl již použit minimálně jeden filtr typu horní propust. Obsahují tedy vysokofrekvenční složky, které nesou informace o detailech obrazu.

Pro rekonstrukci takto vzniklých koeficientů a obnovení původního obrazu jsou opět koeficienty decimovány a dále jsou použity rekonstrukční filtry. Uvedený proces je inverzní vlnková transformace (IDWT2).



Obr. 9: DWT2 - zpracování po řádcích a po sloupcích

### 2.4.7 Kvantování

Jedná se o proces, který umožňuje za cenu snížení přesnosti koeficientů DWT dosáhnout vyšší komprese [9]. Při procesu kvantování dochází ke ztrátě dat. Aplikuje se zde kvantování označované jako Embedded Deadzone Quantization [7] – kvantování s mrtvou zónou DZQ. Jedná se o skalární kvantování. Kvantizační krok lze volit odděleně pro jednotlivé úrovně DWT, a tím do jisté míry ovlivňovat kvalitu. Je závislý na dynamickém rozsahu příslušného subpásma a na počtu bitů reprezentujících původní obrazovou složku. Při použití LeGall filtru 5/3 se kvantizace neprovádí. Jedná se o filtr využívaný výhradně při bezztrátové kompresi. Při přípravě pro entropické kódování dojde k rozdělení na nepřekrývající se čtvercové oblasti, které tvoří v obraze přes subpásma bloky o stejné velikosti.

### 2.4.8 Entropické kódování a tvorba souboru dat

Poslední činnost, která se snaží eliminovat nadbytečnost, je aritmetické entropické kódování. Do tohoto bloku komprimačního řetězce vstupují bloky koeficientů DWT a na výstupu je komprimovaný datový tok. V algoritmu JPEG 2000 byla použita metoda Embedded Block Coding with Optimal Truncation – blokový kodér s optimálním zkrácením EBCOT [3]. Tato část komprese je bezztrátová.

Jedná se o zakódování veškerých koeficientů v bloku, při kterém se postupuje od nejvýznamnějšího po nejméně významný. Dojde k rozdělení koeficientů do tzv. bitových rovin, které se poté zpracovávají samostatně. U těch, které obsahují nulu, dojde k zapamatování jejich počtu a přeskočení. Načítání probíhá po čtyřřádkových blocích v rámci roviny a po čtyřbitových sloupcích v rámci bloku.

U této metody se využívá tří typů průchodů:

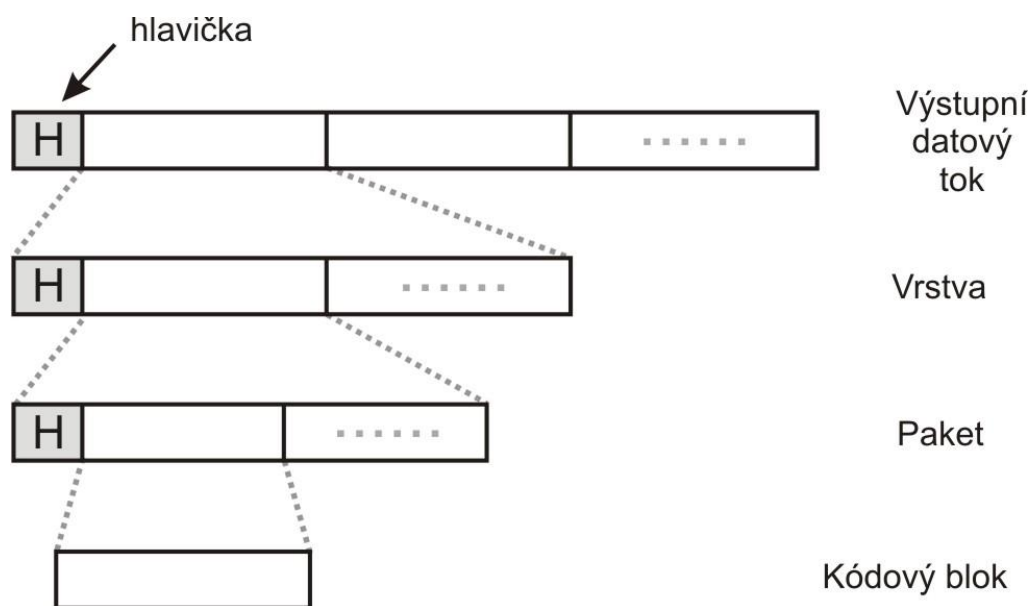
**Significant** - kódují se bity nevýznamné, mající významné sousedící bity (alespoň jeden z nich v jednom z osmi směrů)

**Magnitude** - kódují se bity, na jejichž pozicích byly v minulých rovinách bity významné

**Cleanup** - kódují se všechny zbylé bity

Všechny roviny jsou kódovány všemi třemi typy průchodů, přičemž prvním je vždy Cleanup. Znaménko je kódováno s prvním bitem. Významnost bitu se hodnotí dle jeho hodnoty: je-li v této nebo předchozí rovině roven 1, je významný.

Tento blok zajišťuje správné pořadí dat v bitsreamu a přidání daných datových hlaviček do výsledných paketů. Hlavičky obsahují informace o bitových rovinách, o množství nulových bitových rovin, o obsazených bitových rovinách a další. Pro řízení kvality u ztrátové komprese lze méně významné pakety vypustit, u bezztrátové k tomu nesmí dojít.



Obr. 10: Struktura datového toku [6]

## 3 ZABEZPEČOVACÍ METODY PRO MOTION JPEG 2000

### 3.1 Motion JPEG 2000

Videoformát Motion JPEG 2000 je definován v ISO/IEC 15444-3, respektive v ITU-T.802 [14]. Konkrétně jej definuje 3. část standardu. Je určen pro rozšíření statických obrázků ve formátu JPEG 2000. Specifikuje použití formátu JPEG 2000 pro časové sekvence snímků (pohybové sekvence), případně jeho kombinaci s audio stopou. Také definuje formát souborů založený na bázi ISO multimediálních souborů. Přípony videosouborů pro Motion JPEG 2000 jsou .mj2 a .mjp2 podle RFC 3745. Na rozdíl od běžných videokodeků jako například MPEG-4, WMV a DivX, Motion JPEG 2000 nepoužívá časovou nebo mezisnímkovou kompresi. Každý snímek je kódován jako nezávislý subjekt, buď ztrátovou, nebo bezztrátovou variantou JPEG 2000 [15]. Pro audio stopu se užívá LPCM kódování (Lineární pulsní kódovou modulaci).

Motion JPEG 2000 nabízí funkce, které z něj dělají dobrou volbu pro ukládání dat ve vědeckých, bezpečnostních a výzkumných oblastech [3]:

- **Bezztrátová kompresní varianta** - ze souboru lze rekonstrukcí získat původní obrazová data
- **Granulární přístup** - animace se může skládat z jednotlivých komponent, dlaždic nebo oblastí snímku
- **Intra-frame kódování** - každý snímek je nezávislý subjekt s reálnou reprezentací dat v jednom okamžiku. Standard MPEG používá Inter-frame kódování, kde je díky vzájemné závislosti mezi snímky nemožné získat jednotlivá snímková data.

Díky principu Intra-frame kódování dochází pouze k odstranění redundance v jednotlivých snímcích. Ve srovnání s jinými moderními kodeky, které jsou určeny výhradně pro kompresi videosekvencí, poskytuje nižší kompresní účinnost. Tyto odstraňují redundanci pomocí podobnosti po sobě jdoucích snímků a dosahují tím lepších výsledků.

## 3.2 Zabezpečení formátu proti přenosovým chybám

Technologický vývoj v oblasti multimediálních technologií jde obrovským tempem vpřed. Masivnější začlenění mobilních zařízení do běžného každodenního života, integrace kamer a fotoaparátů s čím dál vyšším rozlišením a stoupající obliba sociálních sítí a multimédií v této oblasti vede k obrovským nárokům na přenosové kanály pro tyto obrazové a zvukové informace. Bezpečné kódování je často nedostatečné díky požadavku na malý objem přenášených dat. JPEG 2000 proto například umožňuje zabezpečovací informace uložit ve zdrojovém souboru a přenášet je pouze je-li to potřeba, čehož je dosaženo strukturou datového toku a nezávislou činností zpracování obrazových dat.

V jednotlivých paketech jsou přenášena data kódových bloků, respektive bitových rovin. Díky jejich zpracování se chyba vyskytuje jen v daném bloku, který určuje poškozenou oblast. Dojde-li k poškození paketu, který obsahuje blok s informacemi celého subpásmá, toto poškození se projeví v celé ploše obrazu. Je to tím, že pro vlnkové koeficienty jednotlivých subpásem *LL*, *LH*, *HL*, *HH* platí, že čím výše v úrovních postupujeme, tím zaujímají bloky větší plochu subpásmá, což je nežádoucí. Poškození je ovlivněno dle konkrétního subpásmá. Největší část informací o obrazu se nachází v subpásmu *LL*, proto jeho poškození bude mít na obraz nejhorší dopad a naopak ve zbývajících subpásmech se chyby projeví s menším důsledkem [9].

Ochraně proti chybám je věnována pozornost ve specifikaci standardu převážně ve dvou částech. Jedná se o základní nástroje v primární 1. části - Core coding systém, dále o rozšiřující nástroje definované v 8. části označené JPSEC [16] a následně rozšířené možnosti ochrany převážně při využití bezdrátových přenosových sítí definované v 11. části, která se zapojila do standardu v roce 2007.

Dalším typem metod jsou ty, které se už nesnaží data zabezpečovat, nýbrž s chybami počítají. Jejich cílem je minimalizovat konečný dopad na kvalitu obrazu. Využívají nepoškozených obrazových dat k rekonstrukci poškozených. Jedná se o maskovací metody. U maskovacích metod aplikovatelných na videosekvenci Motion JPEG 2000 lze použít dvou způsobů: metody LL subband a metody filtrační.

Uvažujeme-li použití první z metod, tedy LL subband, využijeme skutečnosti, že ve videosekvenci dochází pouze k malé časové změně mezi snímky, a ty si zachovávají jistou podobnost. Čím je tato

podobnost větší, je účinnost této metody vyšší. Naopak vykazují-li po sobě jdoucí snímky jen malou shodu, například u velmi dynamických sekvencí, může být tato metoda méně účinná.

U filtračních metod dochází k rekonstrukci pomocí nepoškozených dat ve snímku samotném. Využívá prostorové filtry, které účelně potlačují dané chyby. Filtrační metody je vhodné doplnit hranovými detektory, které zajistí obnovu hran. Metody lze také kombinovat.

Potřebujeme-li porovnávat druhy komprese mezi sebou a srovnávat dopady vzniklých chyb nebo aplikace některých z maskovacích metod, užívá se pro posouzení kvality obrazu nebo jeho kvality po rekonstrukci měření vizuální kvality komprese a její ztrátovosti [17].

### 3.3 Posuzování vizuální kvality komprese

Zkomprimujeme-li obraz sebelepší kompresí, obraz se vždy bude lišit od obrazu původního, originálního. Existují proto metody subjektivního a objektivního hodnocení kvality. Všechny objektivní metody mají na rozdíl od subjektivních přesně dané parametry, které definují vztahy mezi měřitelnými hodnotami.

Jedna z objektivních metod, kterou lze měřit ztrátovost komprese, je střední kvadratická chyba MSE (Mean Square Error). Jde o zkreslení zkomprimovaného obrazu, který je porovnáván s originálem [18].

Označíme-li původní nezpracovaný snímek  $X$ , tedy výchozí posloupnost bude  $X = \{x_1, \dots, x_n\}$ , a komprimovaný obraz po průchodu dekodérem a po rekonstrukci označíme  $Y$ , tedy výsledná posloupnost je  $Y = \{y'_1, \dots, y'_n\}$ , potom platí pro MSE následující vztah:

$$MSE = \frac{1}{n} \sum_{i=1}^n (x_i - y'_i)^2$$

kde  $N$  představuje počet obrazových bodů ve snímku.

Další takovou metodou k měření ztrátovosti komprese je poměr signálu k šumu SNR (Signal to Noise Ratio):

$$SNR = 10 \log_{10} \frac{P^2}{MSE^2} [dB]$$

kde  $P$  udává interval hodnot vstupní posloupnosti  $P = \max - \min$ .

Nejčastěji používanou objektivní metodou je špičkový poměr signálů k šumu PSNR (Peak-Signal to Noise Ratio), který je vyjádřen z první uváděné metody střední kvadratické chyby. V našem případě posuzování kvality rekonstruovaných obrazů jde o posouzení rozdílu mezi bezchybným originálním a zrekonstruovaným obrazem [17].

Definice metody je následující:

$$PSNR = 10 \log_{10} \frac{M^2}{MSE^2} [dB]$$

kde M udává maximum intervalu hodnot vstupní posloupnosti, například pro vyjádření hodnoty pixelu pomocí 8 bitů je M=255.

Tyto uváděné objektivní kritéria neodpovídají subjektivním. Ty by ale měly být primární, jelikož zpravidla je posuzovaný obraz určen lidem. Existují ovšem případy, například videosekvence, při kterých se díky rychlé časové posloupnosti, a tím pádem velkému množství zpracovaných obrazů, nehodí použít subjektivní metody a měla by být užita například metoda PSNR.

Při subjektivním hodnocení jsou jedním z nejdůležitějších činitelů při určování kvality hrany. Pro lidské vnímání působí obraz s rozmazanými hranami nekvalitně, a to se na tomto způsobu hodnocení kvality projeví. Výsledkem jsou slovní hodnocení nebo číselná známka.

## 3.4 Implementované zabezpečovací mechanizmy

### 3.4.1 Part 1 – Core Coding Systém

V první části standardu JPEG 2000 jsou specifikovány základní možnosti ochrany proti chybám navazující na proces tvorby kódových bloků jako výstup kompresního algoritmu. Vznikne-li porucha při přenosu v části tohoto bloku, dojde k chybnému vnitřnímu stavu dekodéru a na jeho výstupu se objeví chybně interpretovaná obrazová data. Nástrojem proti tomuto stavu je reinicializace dekodéru po jednotlivých blocích. Další možností je užití pomocných značek paketů, označujících začátek paketu - SOP (Start Of Packet) a konec paketu – EPH (End of Packet Header) [13], [2]. Ty umožňují restart dekodéru na hranici sub-bitové vrstvy a omezení projevu těchto chyb na malou část výstupního obrazu.



Uspořádání do jednotlivých kvalitativních vrstev umožňujících například postupné načítání s rostoucím rozlišením, zvyšuje citlivost datového toku. Zvyšuje se tím množství dat v hlavičkách a dochází i ke snížení kompresního poměru. Možné poškození hlavičky paketu má za následek ztrátu celého paketu. Proto je v tomto případě vhodnější použít co největší členění do většího počtu paketů, díky čemuž dojde ke znehodnocení menší části dat.

### 3.4.2 Part 8 – JPSEC

V první části standardu nejsou specifikovány žádné zabezpečovací funkce například pro autentizaci nebo podmíněný přístup. Právě tyto možnosti nabízí 8. část s názvem Secure JPEG 2000 – JPSEC. Ten poskytuje zabezpečení pomocí autentizace obrazu, bezpečné škálovatelné překódování pro distribuci obrazu i videa a podmíněný přístup k danému obsahu. Jedná se o standard ISO/IEC 15444-8 [3].

Všechny nástroje, které JPSEC obsahuje, musí být registrovány registrační autoritou JPSEC. Jsou tak označeny jedinečným identifikátorem, který slouží k získání detailních informací k tomuto nástroji formou dotazu na RA (Registration Authority) [16]. Touto formou lze v budoucnu registrovat a aplikovat další zabezpečovací metody.

#### Poskytované nástroje:

- **Podmíněný přístup** – šifrování obrazových dat nebo jeho částí a následný souhlas nebo nesouhlas s přístupem k těmto datům. Využití je možné v placené obrazové nebo video galerii, kdy lze požadovaný obsah zobrazit pouze v malém prostorovém rozlišení.
- **Autentizace** – ověřování identity uživatele nebo zdroje dat. V tomto případě je užito běžných kryptografických metod – digitální podpis, vodoznak, hašování.
- **Bezpečné škálovatelné překódování** – síťový uzel disponuje možností streamovat nebo překódovat JPSEC, aniž by bylo potřeba dešifrovat celý obsah

#### JPSEC syntaxe

V syntaxi je užito značek SEC (poskytuje informace o aplikovaných metodách a o oblastech jejich aplikace) a INSEC (určuje parametry k použité zabezpečovací metodě uvedené v SEC). Značkovací segment SEC je povinně umístěn v hlavičce. INSEC je nepovinný segment a lze jej umístit kdekoli v datovém toku [16].



Obr. 11: Syntaxe segmentové značky SEC [16]

SEC – označuje začátek segmentu (0xFF94)

$L_{SEC}$  – určuje délku segmentu

Tool<sub>i</sub> – obsahuje informace o metodě i



Obr. 12: Syntaxe segmentové značky INSEC [16]

INSEC – označuje začátek segmentu (0xFF95)

$L_{INSEC}$  – určuje délku segmentu

i – index identifikující ochrannou metodu i

R – určuje, zda-li je segment určen pro předchozí nebo následující datový blok

AP – obsahuje doplňkové informace k metodě i

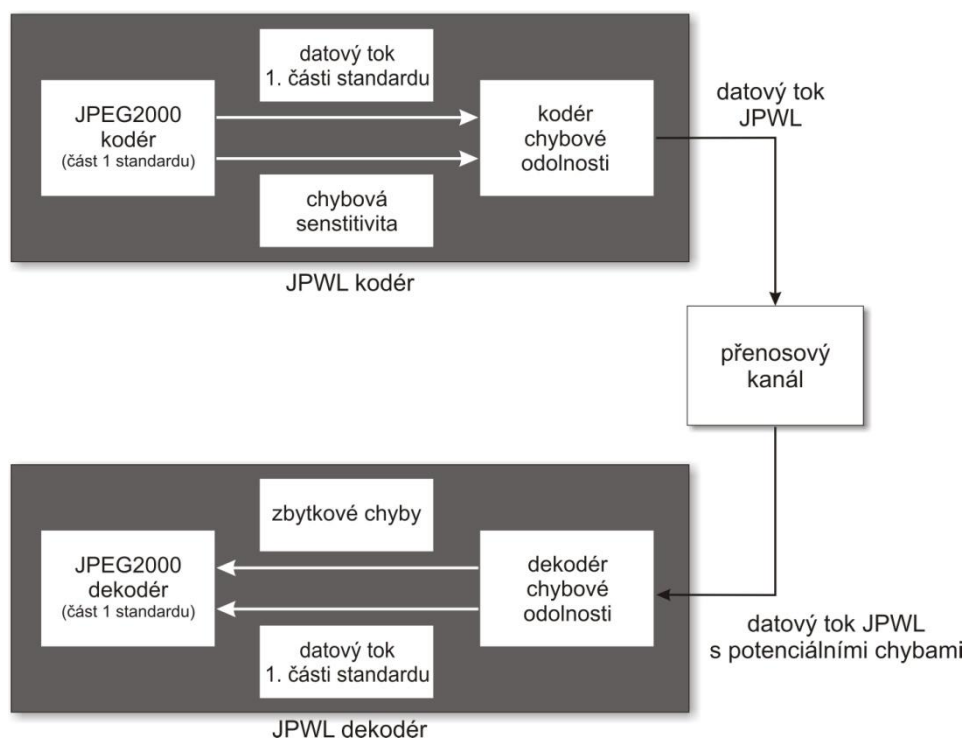
### 3.4.3 Part 11 – JPWL

Má-li být JPEG 2000 vhodný pro bezdrátové multimediální aplikace, musí být také odolný proti charakteru chyb, který je pro tyto sítě typický. V předchozích kapitolách byly popsány nástroje k vytvoření chybové odolnosti, ať už reinicializace dekodéru nebo metody JPSEC, které nezajišťují dostatečné zabezpečení při přenosu po bezdrátových přenosových kanálech. Pro řešení tohoto problému je ve standardu obsažena část 11, definující JPWL (JPEG 2000 Wireless) [19]. Ta rozšiřuje možnosti popsané v části 1 o další metody vhodné pro bezdrátové sítě. Prioritní novou metodou je možnost označit citlivost jednotlivých datových částí na výskyt chyby. To umožňuje tyto citlivější data zabezpečit lépe než jiné (nerovnoměrná chybová ochrana), čímž je zvýšena úroveň zabezpečení, ale zároveň nevznikají zbytečná redundantní data.

**Systém JPWL podporuje 3 následující metody [3]:**

- Ochrana datového toku proti chybám
- Popis úrovně citlivosti různých částí datového toku na přenosové chyby
- Popis umístění zbytkové chyby v datovém toku

JPWL systém je ilustrován na následujícím obrázku:



Obr. 13: Schéma JPWL systému [3]

Jak je z obrázku patrné, data generovaná kódérem JPEG 2000 jsou odlišně označena dle úrovně jejich chybové citlivosti (error sensitivity) a díky tomu lze aplikovat odpovídající metodu pro ochranu. Pokud se po průchodu poruchovým přenosovým kanálem nepodaří vzniklé chyby odstranit, jsou opět odpovídajícím způsobem označeny (zbytkové chyby). To později poskytuje dekodéru volbu jak s těmito daty naložit.

Struktura JPEG 2000 umožňuje snadné zavedení 4 nových značek použitých v JPWL [19]. Neprojeví se přitom problémy s dekodéry, které s těmito značkami nepracují. Jsou schopny

obrazová data s těmito značkami správně dekomprimovat pomocí jejich vynechání. Tím je zajištěna zpětná kompatibilita.

### **EPC – Error Protecting Capability**

Schopnost chybové ochrany říká, jaké principy ochrany byly v datovém toku použity. Určuje, jaké metody bylo použito k vytvoření ochranných informací, které jsou zapotřebí ke zpětné obnově.

### **EPB – Error Protection Block**

Blok chybové ochrany zajišťuje ochranu záhlaví paketů. Těm nebyla v 1. části standardu věnována žádná zabezpečovací metoda. Tato značka poskytuje blok chráněných dat za cenu přidání redundantních dat pro potlačení chyb. Při obousměrné komunikaci může být tento blok vyslán opakovaně jako odpověď poptávací straně z důvodu velkého poškození.

### **ESD – Error Sensitivity Descriptor**

Popisovač chybové citlivosti nabízí informace o tom, jakou má aktuální datový blok informační váhu při možném vzniku chyby. Poskytuje identifikaci rizikových oblastí v datech.

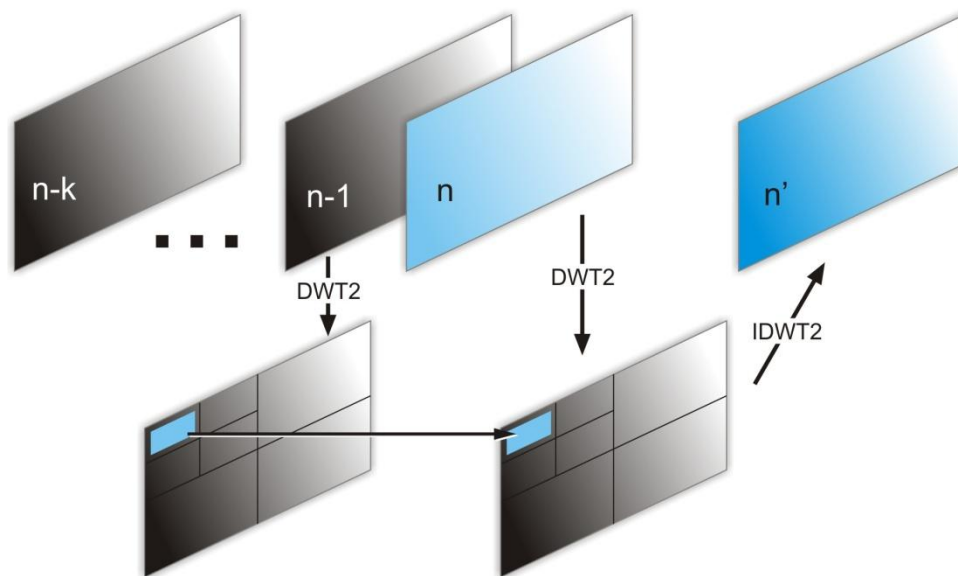
### **RED – Residual Error Descriptor**

Popisovač zbytkových chyb je využit, jsou-li použité zabezpečovací metody nebo datová redundance nedostatečné, a tím nedojde k opravě chyb. Tyto jsou označeny jako zbytkové chyby a poskytují dekodéru možnost volby jak tento poškozený blok správně interpretovat a tím zabránit poškození.

## **3.5 Metoda LL Subband**

Při použití této metody dojde k odstranění LL subpásma poškozeného snímku a k jeho kompletnímu nahrazení LL subpásmem ze snímku opraveného. Je tedy založena na nahrazování LL subpásma. Chceme-li tuto metodu využít, budeme potřebovat zachovat LL subpásma ze správných snímků neovlivněných přenosovými chybami. K tomu nejvhodněji poslouží předchozí snímek před snímkem poškozeným díky jeho nejbližší shodě. Budeme-li tedy nahrazovat LL subpásma u snímku  $n$ , budeme k tomu potřebovat  $n-k$  uchovaných snímků. Uvažujeme-li díky nejbližší shodě použití předcházejícího snímku, jedná se o  $n-1$ .

Přijmeme-li tedy poškozený snímek  $n$ , tak následně aplikujeme DWT2 jak na snímek  $n$ , tak na snímek předchozí  $n-1$ . Následuje záměna LL subpásma snímku  $n$  na subpásma získané ze snímku  $n-1$ . K obnovení opraveného snímku se poté aplikuje IDWT2 [9].



Obr. 14: Princip metody LL subband [9]

LL subpásma neobsahuje informace o detailech v obraze. Díky tomu jsou v obraze zachovány jeho detaily i při výměně tohoto prvku. Jejich uchování je úměrné úrovni DWT2. Čím vyšší je tato úroveň, tím je podobnost s původním obrazem vyšší. Po odstranění LL subpásma jsou v obraze zachovány převážně místa, kde se nacházejí hrany, jak ukazuje první část obrázku 15. Druhá prezentuje výsledek po nahrazení LL subpásma poškozeného obrazu.



Obr. 15: Ukázka aplikace metody LL subband [9]

### 3.6 Zajištění datového toku Motion JPEG 2000

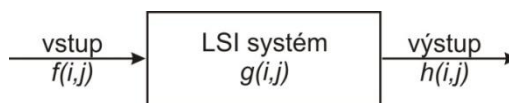
Jak bylo uvedeno, formát Motion JPEG 2000 není nic jiného než sled obrazů komprimovaných pomocí JPEG 2000. Dojde-li v takovém datovém toku k poškození hlavičky souboru u jediného obrazu, dojde k narušení videosekvence, jelikož právě v tomto místě datového bloku jednotlivých obrazů jsou umístěny informace pro dekompresi. V takovém případě snímek nelze zobrazit. Jako možnost zachování spojitosti videosekvence se v tomto krajním případě nabízí možnost nahrazení celého takto poškozeného snímku snímkem předchozím a zajistit tak kontinuitu. Jedná se v podstatě o doplnění maskovací metody LL subband [9].

### 3.7 Filtrační metody

Filtrační metody se používají pro obnovení respektive opravu chyb vzniklých vlivem šumu nebo jiných nežádoucích veličin působících na formát Motion JPEG 2000 při všech fázích přenosu.

#### 3.7.1 Lineární filtry

Lineární filtrační metody určují novou hodnotu zpracovávaného pixelu jako lineární kombinaci hodnot pixelů z jeho zvoleného okolí. Filtrace může být modelována prostorově invariantním lineárním systémem LSI (Linear Space Invariant System) [20]. Prostorově invariantní je systém, v našem případě filtr, který vykazuje stejné vlastnosti v libovolném bodě obrazu. Průběh výpočtu bude pro každý bod obrazu stejný.



Obr. 16: LSI systém

Jak vidíme na blokovém schématu, LSI systém je popsán jako impulsní odezva  $g(i, j)$ .

Má-li být systém lineární, musí splňovat podmínku:

$$a \cdot f_1(i, j) + b \cdot f_2(i, j) \rightarrow a \cdot h_1(i, j) + b \cdot h_2(i, j)$$

Kde  $f_1(i, j)$  a  $f_2(i, j)$  jsou vstupní obrazy,  $h_1(i, j)$  a  $h_2(i, j)$  jsou obrazy odpovídající odezvě filtru, a a b jsou konstanty [21].

Výstup takového systému označený  $h(i, j)$ , který splňuje výše uvedenou podmínku, je dán konvolucí vstupního obrazu  $f(k, l)$  a impulsní odezvy  $g(i, j)$ .

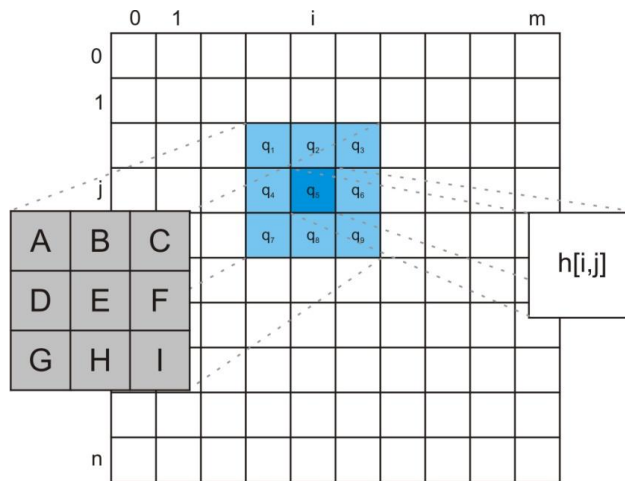
$$h(i, j) = f(i, j) * g(i, j) = \sum_{k=-\frac{n}{2}}^{\frac{n}{2}} \sum_{l=-\frac{m}{2}}^{\frac{m}{2}} f(k, l) g(i - k, j - l)$$

V oblasti zpracování obrazu představuje konvoluce součet vážených sum pixelů v obrazu. Impulsní odezva  $g[i, j]$  je označována jako konvoluční maska. Jednotlivý pixel obrazu  $[i, j]$  je potom použit pro stanovení hodnoty výstupního pixelu tím, že střed konvoluční masky se posune do bodu  $[i, j]$  a určí se vážená suma pixelů v okolí  $[i, j]$ , přičemž hodnotám v konvoluční masce odpovídají jednotlivé váhy pixelů [20].

Hodnota na výstupu se tedy určí jako vážená suma

$$h[i, j] = Aq_1 + Bq_2 + Cq_3 + Dq_4 + Eq_5 + Fq_6 + Gq_7 + Hq_8 + Iq_9$$

kde A,B,C...I jsou váhy odpovídající hodnotám  $g[i, j]$



Obr. 17: Určení odezvy  $h[i, j]$  s konvoluční maskou 3x3 se středem konvoluční masky odpovídající hodnotě E [20]

### 3.7.2 Průměrovací Filtr (Mean Filter)

Jedná se o jeden z nejjednodušších lineárních filtrů, realizující operaci lokálního průměru. Průměr ze všech hodnot pixelů z okolí pixelu ve vstupním obraze nahradí hodnotu tohoto pixelu ve výstupním obraze. Průměrovací filtr, počítající s oknem 3x3, je dán následujícím vztahem:

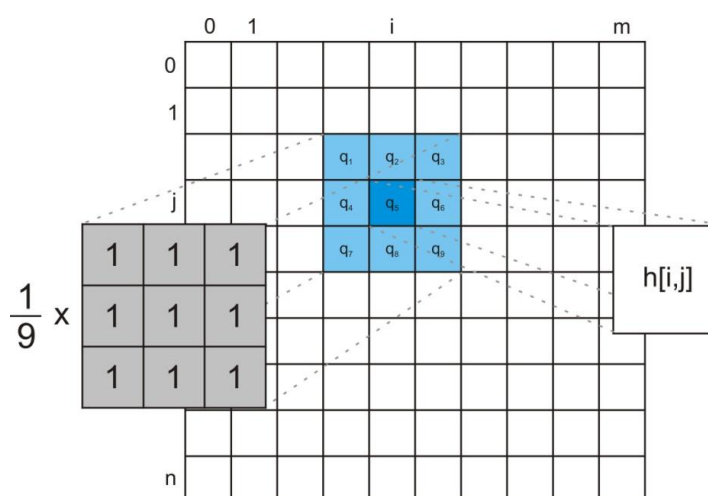
$$h[i,j] = \frac{1}{9} \sum_{k=i-1}^{i+1} \sum_{l=j-1}^{j+1} f[k,l]$$

Odpovídající konvoluční maska má následující tvar:

$$h = \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Došlo ke zredukovaní konvoluce na operaci lokálního průměru, jehož hodnotu určuje průměrná hodnota z daného okolí. Je to způsobeno tím, že v porovnání s rovnicí pro výstup systému LSI uvedenou výše, mají všechny hodnoty konvoluční masky stejnou hodnotu 1/9 [21].

Velikost okna má značný vliv na kvalitu výstupu takového filtru. Bude-li zvolené okno moc malé, k žádnému výraznému zkvalitnění nemusí dojít. Naopak zvolíme-li okno moc velké, kvalita výsledného obrazu bude trpět značnou rozostřeností. Ovšem i vhodně zvolenou velikostí okna k určitému rozmazání hran v obraze dojde, proto je tento jednoduchý filtr většinou používán pouze jako pomocný v pokročilejších nelineárních filtračních metodách.



Obr. 18: Průměrovací filtr se svou konkrétní konvoluční maskou 3x3 [20]

### 3.7.3 Kruhový filtr

Jeho funkce je stejná jako u filtru průměrovacího pouze za použití jiného tvaru konvoluční masky. V té je změněna váha jednotlivých prvků, přičemž kolem středního prvku je váha jednotlivých



kruhových prvků vyšší. Ke zvýšení váhy, například při užití masky 3x3, dochází u 4 jeho sousedů. Může mít následující tvar:

$$h = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

### 3.7.4 Rekurzivní filtry

Důležitou skupinou prostorově invariantních filtrů jsou rekurentní filtry. Při aplikaci běžné konvoluce se zjištěné hodnoty při posunu konvoluční masky uloží rovnou do výstupního obrazu a pro další výpočet se využijí původní nezměněné hodnoty. Použijeme-li jako vstupní hodnoty pro konvoluci hodnoty zjištěné v předchozím kroku a v předchozí poloze masky, jedná se o rekurzivní filtr [21].

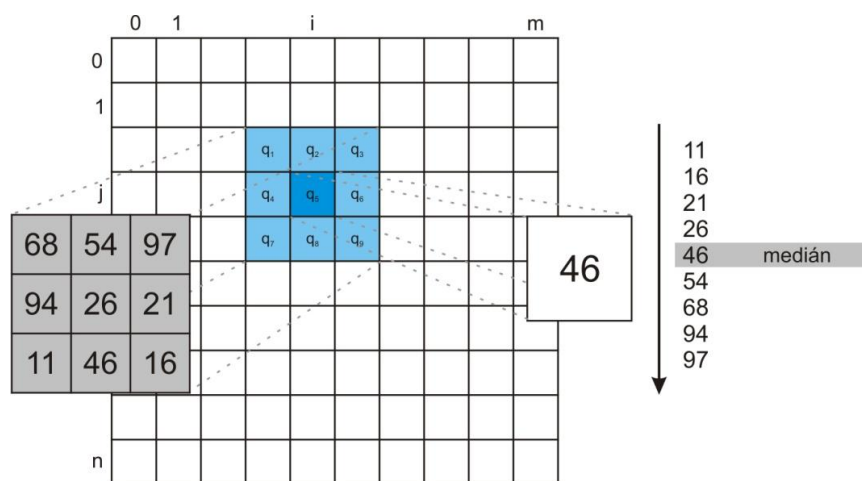
### 3.7.5 Filtrační metody – nelineární

Nepředstavuje-li filtr váženou sumu okolních pixelů, potom nesplňuje podmínku linearitu a jedná se o nelineární filtr. Právě tento druh filtrů částečně řeší problém s rozmazáváním hran u filtrů lineárních. I tento typ analyzuje okolní oblast za účelem nalezení okolí s co nejvíce konstantním jasem. Tato oblast je poté použita při určování výsledné jasové hodnoty.

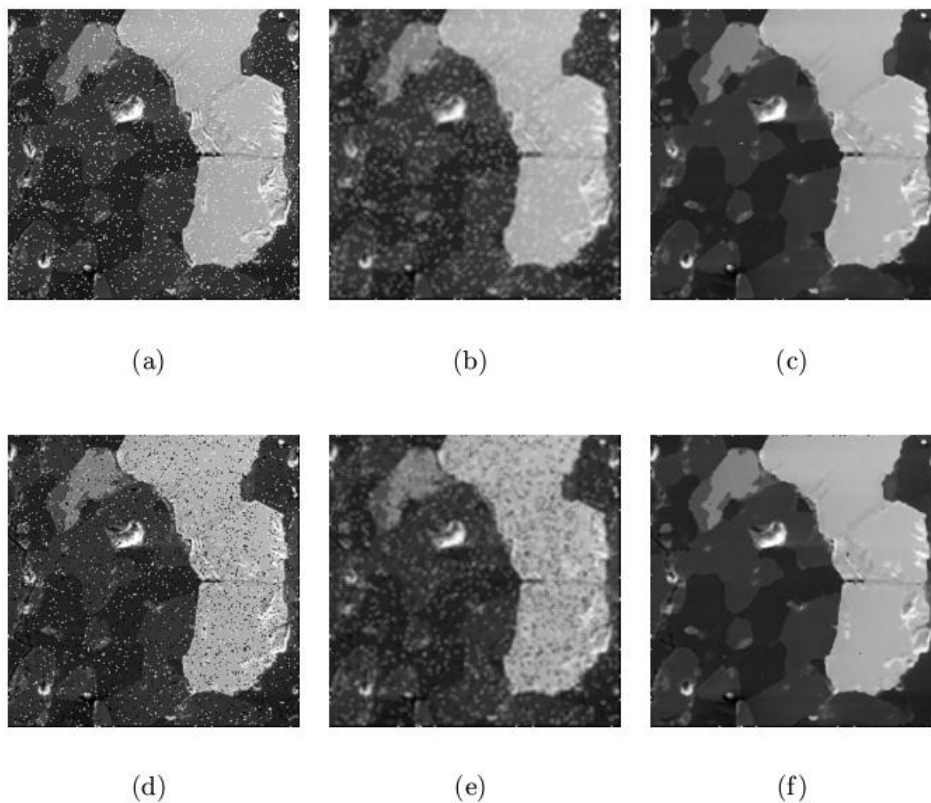
### 3.7.6 Mediánový filtr

Tento filtr patří do skupiny nelineárních filtrů. Splňuje ale podmínky prostorové invariance. V mediánovém filtru je hodnota každého pixelu nahrazena hodnotou mediánu určeného z pixelů v jeho okolí. Medián se získá tak, že po vzestupném seřazení jednotlivých pixelů dle jejich hodnoty vybereme ten, který se nachází uprostřed. To platí pro lichý počet pixelů. Je-li jejich počet sudý, vybereme dva střední prvky a jako výstup označíme jejich průměr [20].

Tato metoda snižuje úroveň rozmazání hran. Výrazně snižuje i impulsní šum. Mediánový filtr lze aplikovat iterativně. Při zvolení obdélníkového okolí pixelů vzniká problém porušení ostrých rohů v obraze a tenkých čar. Ten lze eliminovat zvolením jiného vhodného tvaru okolí.



Obr. 19: Mediánový filtr se svou konkrétní konvoluční maskou 3x3 [20]

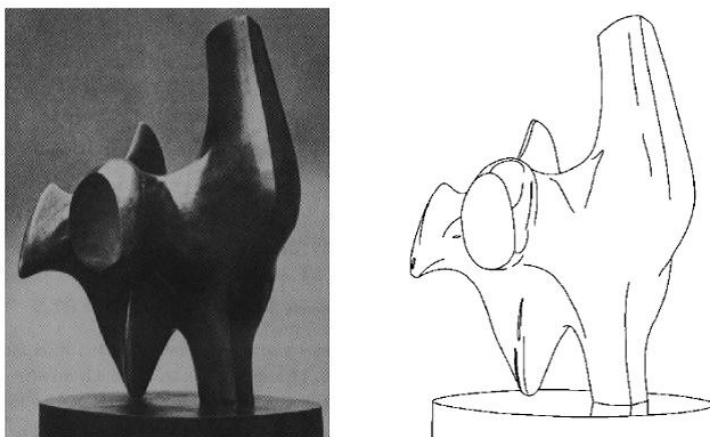


Obr. 20: Náhled výsledku průměrovacího filtru a mediánového filtru [20]  
a) obraz poškozený impulsním šumem, d) poškozený šumem typu sůl a pepř  
b) a e) průměrovací filtr, c) a f) mediánový filtr

### 3.8 Detekce Hran

Při aplikaci filtračních metod z předešlé kapitoly více či méně dochází k degradaci hran v obraze. Díky neurofyzilogickým a psychologickým výzkumům se ukázalo, že téměř nejdůležitějšími místy v obraze pro přirozené lidské vnímání jsou právě hrany. Hranami označujeme oblasti v obraze, ve kterých se prudce mění hodnota jasu. Proto je pro udržení nebo zlepšení subjektivní kvality obrazu snaha tyto hrany zachovávat. K tomu je zapotřebí detekce hran neboli hranové detektory.

Řekněme, že chceme použít filtrační metodu k potlačení některých z nežádoucích vzniklých chyb. Aplikujeme-li ještě před vlastní filtrací hranové detektory k zachování hran v obraze, a po filtraci dojde k opětovnému obnovení hran, obnovíme tím ostrost jinak sníženou vlivem filtračních metod.



Obr. 21: Ilustrace detekce hran

#### 3.8.1 Gradientní operátory

Hrana v obraze je určena vlastnostmi obrazového elementu – pixelu a jeho okolí. Určuje ji to, jak prudce se mění hodnota obrazové funkce  $f(x, y)$ . Nástrojem, který nám umožňuje studium změny funkce dvou proměnných, jsou parciální derivace [21]. Samotnou změnu funkce nám určuje její gradient. Jde o vektorovou funkci. Směr gradientu určuje směr největšího růstu funkce a velikost, nebo-li modul gradientu, udává strmost růstu funkce. Velký modul gradientu jako vlastnost pixelu potom ukazuje, že se jedná o hranu.

Gradient spojitě funkce  $n$  proměnných je vektor parciálních derivací [22]:

$$\nabla f(x_1, \dots, x_n) = \left( \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right)$$

Při  $n = 1$  je roven obyčejné derivaci

Při  $n = 2$  - obrazová funkce  $f(x, y)$ , velikost gradientu je dána vztahem [21]:

$$|\nabla f(x, y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Směr gradientu udává vztah:

$$\phi = \arg\left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$

kde  $\arg(\cdot)$  je úhel mezi souřadnou osou  $x$  a vektorem argumentu funkce v radiánech.

Gradientních operátorů se využívá i při ostření obrazu. Ostření obrazu znamená upravit oblast v obraze tak, že v něm budou strmější, tím pádem ostřejší a výraznější hrany. Ostření obrazu prakticky odpovídá zdůraznění vysokých frekvencí. Je-li obraz  $f$  výsledkem ostření obrazu  $g$ , můžeme napsat:

$$f(x, y) = g(x, y) - CS(x, y)$$

kde  $C$  je součinitel roven použité síle ostření a  $S$  udává strmost změny funkce v daném pixelu. Strmost je dána modulem gradientu [21].

Parciální derivace se v obraze aproximuje následovně:

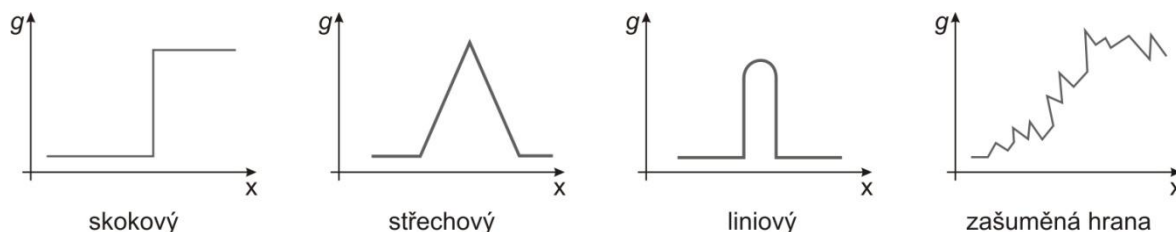
$$\Delta_x f(x, y) = f(x + 1, y) - f(x, y)$$

$$\Delta_y f(x, y) = f(x, y + 1) - f(x, y)$$

Vztahy lze vyjádřit i symetricky, ale toho se nevyužívá, protože tato aproximace nezohledňuje vliv samotného pixelu  $(x, y)$ .

### **Jasové profily v okolí hran**

Na prvních třech grafech v následujícím obrázku 22 jsou uvedeny nejprve teoretické průběhy jasu kolem hranových bodů. V posledním grafu je reálný zašuměný jasový profil hrany [22].



Obr. 22: Jasové profily [22]

**Hranové detektory, respektive gradientní operátory, lze rozdělit do tří kategorií:**

- Detektory založené na hledání maxima prvních derivací (Roberts, Prewittová, Sobel)
- Operátory založené na hledání hran v oblastech, ve kterých druhá derivace obrazové funkce prochází nulou - zero-crossing, příkladem takového detektoru je Marr-Hildreth a LoG
- Detektory založené na lokální aproximaci obrazové funkce jednoduchými parametrickými modely, příkladem je polynom dvou proměnných

### 3.8.2 Hranové detektory

Zvýraznění hran v obraze způsobí aplikace dané konvoluční masky hranového detektoru při filtraci obrazu. Pro relevantní výsledek je po filtraci gradientním operátorem aplikováno prahování. Při tom se pomocí rozhodovací úrovně (prahu) rozhoduje, zda je výsledek z předchozí filtrace ignorován nebo je uznán jako hrana. Práh, určený pro tento krok, lze zvolit jednotný pro celý obraz nebo jej lze určovat proměnlivě po blocích.

Typy užívaných hranových detektorů:

- Robertsův operátor
- Operátor Prewittové
- Sobelův operátor
- Robinsonův operátor
- Kirschův operátor
- Laplaceův operátor

- Logaritmický detektor
- Cannyho hranový detektor

### 3.8.3 Robertsův operátor

Jedná se o jednoduchý operátor, používající jen okolí 2x2.

Jeho konvoluční masky jsou následující:

$$h_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad h_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Velikost gradientu se zjišťuje dle vztahu:

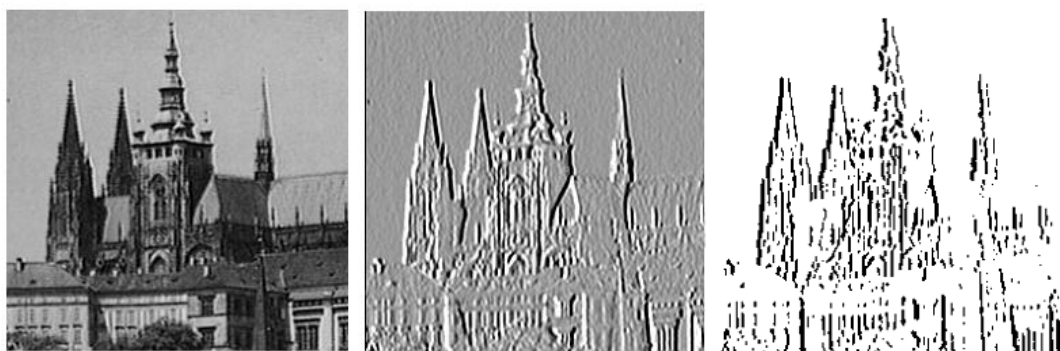
$$|\nabla g(x, y)| = |g(x + 1, y + 1) - g(x, y)| + |g(x, y + 1) - g(x + 1, y)|$$

Nevýhodou tohoto operátoru je větší citlivost na šum, jelikož okolí použité pro aproximaci je pouze 2x2 [22].

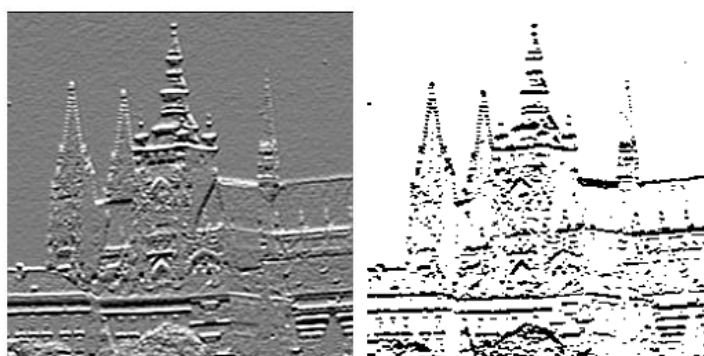
### 3.8.4 Operátor Prewittové

Operátor Prewittové aproximuje první derivaci a jeho gradient je použit z okolí 3x3. Konvoluční maska, pro kterou je určen největší modul gradientu, je použita. Základních 8 konvolučních masek má tvar [21]:

$$\begin{aligned} h_1 &= \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix} & h_2 &= \begin{pmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{pmatrix} & h_3 &= \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \\ h_4 &= \begin{pmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} & h_5 &= \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} & h_6 &= \begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix} \\ h_7 &= \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix} & h_8 &= \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{pmatrix} \end{aligned}$$



Obr. 23: Hrany v severním směru – originální obraz, severní hrany, prahované hrany [22]

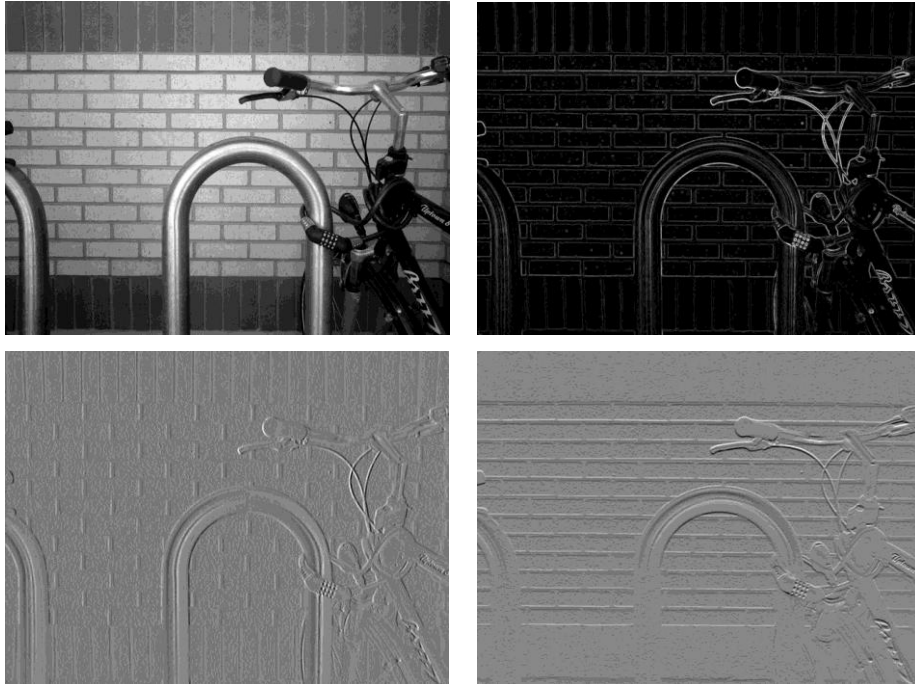


Obr. 24: Hrany ve východním směru – východní hrany, prahované hrany [22]

### 3.8.5 Sobelův operátor

Tento operátor se užívá nejčastěji při potřebě detekce vodorovných a svislých hran [23]. Pak jsou základem dvě konvoluční masky pro filtraci obrazu v horizontálním a vertikálním směru -  $h_1, h_3$ .

$$h_1 = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \qquad h_3 = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$



Obr. 25: Sobelův detektor - originální obrázek, kompletní filtrace, vertikální filtrace, horizontální filtrace

### 3.8.6 Robinsonův operátor

Uvedeny jsou první 3 konvoluční masky. Pootočením jde snadno získat další masky [22].

$$h_1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{pmatrix} \quad h_2 = \begin{pmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{pmatrix} \quad h_3 = \begin{pmatrix} -1 & 1 & 1 \\ -1 & 2 & 1 \\ -1 & 1 & 1 \end{pmatrix}$$

### 3.8.7 Kirschův operátor

Opět jsou uvedeny vztahy pro první 3 masky. Pootočením jde snadno získat další masky [23].

$$h_1 = \begin{pmatrix} 3 & 3 & 3 \\ 3 & 0 & 3 \\ -5 & -5 & -5 \end{pmatrix} \quad h_2 = \begin{pmatrix} 3 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & -5 & 3 \end{pmatrix} \quad h_3 = \begin{pmatrix} -5 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & 3 & 3 \end{pmatrix}$$

### 3.8.8 Laplaceův operátor

Laplaceův operátor je skalární veličina  $\nabla^2 f$ . Při porovnání s předchozími neobsahuje směr hrany. Je invariantní vůči otočení soustavy a představuje jen velikost hrany. Tento operátor je velice



citlivý na šum díky aproximaci druhé derivace [21]. Při tenkých hranách v obraze se u něj projevují nežádoucí dvojité odezvy.

Diskrétní druhá derivace je konvoluce prvních derivací:

$$\frac{d^2}{dx^2} \approx [-1 \quad 1] * [-1 \quad 1] = [1 \quad -2 \quad 1]$$

Konvoluční masky jsou potom definovány takto:

$$h_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad h_2 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Případně lze použít další tvary se zvýrazněním středu, tedy s větší vahou blízkého pixelu:

$$h_3 = \begin{pmatrix} 2 & -1 & 2 \\ -1 & -4 & -1 \\ 2 & -1 & 2 \end{pmatrix} \quad h_4 = \begin{pmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{pmatrix}$$

Potom již není invariantní vůči natočení soustavy [22].

Laplaceův operátor byl vybrán jako příklad pro uvedení průběhu funkce hranových detektorů. Níže je ukázka skriptu a jeho výsledků pro aplikaci v prostředí Matlab.

<code>O=imread('obr.jpg');</code>	<i>načtení obrazu</i>
<code>imshow(O)</code>	<i>zobrazení obrazu</i>
<code>h1 = fspecial('laplacian',0);</code>	<i>inicializace operátoru</i>
<code>lapl1 = imfilter(O,h1);</code>	<i>filtrace maskou h<sub>1</sub></i>
<code>figure(2), imshow(lapl1)</code>	<i>zobrazení výsledku</i>
<code>h2=[1,1,1;1,-8,1;1,1,1];</code>	<i>inicializace operátoru</i>
<code>lapl2 = imfilter(O,h2);</code>	<i>filtrace maskou h<sub>2</sub></i>
<code>figure(3), imshow(lapl2)</code>	<i>zobrazení výsledku</i>
<code>h3=[2,-1,2;-1,-4,-1;2,-1,2];</code>	<i>inicializace operátoru</i>
<code>lapl3 = imfilter(O,h3);</code>	<i>filtrace maskou h<sub>3</sub></i>
<code>figure(4), imshow(lapl3)</code>	<i>zobrazení výsledku</i>
<code>h4=[-1,2,-1;2,-4,2;-1,2,-1];</code>	<i>inicializace operátoru</i>
<code>lapl4 = imfilter(O,h4);</code>	<i>filtrace maskou h<sub>4</sub></i>
<code>figure(5), imshow(lapl4)</code>	<i>zobrazení výsledku</i>



Obr. 26: Původní obrázek, filtrace operátorem  $h_1$ , filtrace operátorem  $h_2$ , filtrace operátorem  $h_3$ , filtrace operátorem  $h_4$

### 3.8.9 Logaritmický operátor

Dle rozdělení hranových detektorů spadá tento do 2. kategorie operátorů založených na hledání hran v oblastech, ve kterých druhá derivace obrazové funkce prochází nulou.

Pro zjištění druhé derivace můžeme využít všesměrový Laplacián  $\nabla_2$ . Tento postup se označuje jako LoG operátor (Laplacian of Gaussian). Díky jeho kombinaci s vyhlazením pomocí Gaussiánu  $G$  je dosaženo ještě větší citlivosti na šum.

Při odvození pomocí postupné první a druhé derivace Gaussiánu dostaneme vztah pro určení hodnot v konvoluční masce [21]:

$$h(x, y) = c \left( \frac{x^2 + y^2 - \sigma^2}{\sigma^4} \right) e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

kde  $x, y$  jsou původní souřadnice,  $c$  je normalizační koeficient který zajišťuje, že součet všech koeficientů v masce je 0.

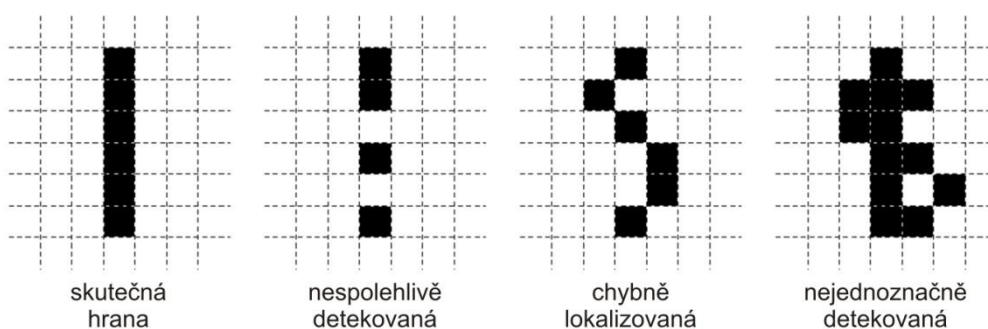
### 3.8.10 Cannyho hranový detektor

Tento algoritmus je považován za nejlepší hranový detektor pro nalezení skokových hran. V dnešní době je běžně používán. Jeho sestavení splňuje tyto základní požadavky:

**Kritérium detekce** – minimální počet chyb – všechny hrany musí být detekovány, nesmí se projevit odezva na místo bez hrany a je nežádoucí vícenásobná odezva na jednu hranu

**Kritérium lokalizace** – přesnost – skutečná hrana se musí od nalezené lišit co nejméně, poloha musí být co nejpřesnější

**Kritérium jediné odezvy** – jednoznačnost – detektor nesmí reagovat na hranu vícenásobně, na rozdíl od prvního kritéria to zajišťuje při značně zašuměných hranách



Obr. 27: Dekódovatelnost hran dle uvedených kritérií [22]

Kroky tohoto detektoru lze rozdělit do čtyř samostatných bloků [21], [23]:

### **Eliminace šumu**

Prvním krokem Cannyho detektoru je aplikace Gaussova filtru pro eliminaci šumu, který je příčinou chybné detekce.

Gaussovo normální rozdělení je dáno vztahem:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$$

kde  $x, y$  jsou souřadnice bodu obrazu a  $\sigma$  je standartní odchylka rozdělení (obvykle 1-1,4).

Vzorcem je určena konvoluční maska, která je následně aplikována na celý obraz.

### **Zjištění gradientu**

Abychom našli velikost a směr gradientů, je použita běžná detekce hrany pomocí vybraného gradientního operátoru. Nejvýhodnější je použít Sobelův operátor, který není citlivý na šum.

### **Zjištění lokálních maxim**

Lokální maxima se zjišťují procesem zvaným ztenčení (thinning). V předchozím kroku jsme získali hodnoty gradientu. Úkolem ztenčení je vybrat jeho lokální maxima. Hrana bude označena v místě s největším gradientem, protože odebereme body, které maximem nejsou. Bodem hrany je označen ten, který má body v okolí kolmém na směr gradientu (známe ho díky Sobelova operátoru) s nižší hodnotou gradientu. Uveďme příklad: máme-li nalezený bod  $X$ , kterým prochází hrana, lze říci, že jeho levý sousedící bod  $X_1$  a pravý sousedící bod  $X_2$  musí mít nižší hodnotu gradientu.

$$X_1 < X > X_2$$

Je-li tomu tak, bod je uznán jako hrana. V opačném případě není.

### **Eliminace bezvýznamných hran**

Tento krok Cannyho hranového detektoru je prahování s použitím hystereze (thresholding). Z předchozího kroku známe přesné místo hran. Nyní posuzujeme její důležitost, protože mohou být označeny i ty nevýznamné. Vícenásobné označení jedné hrany, které může být

způsobeno šumem, je v detekci hran známý problém. Proto jsou běžně výstupy detektorů prahovány a tím rozlišeny hrany významné a nevýznamné. Tímto postupem vznikají hrany, které jsou nesouvislé. Právě tento problém řeší prahování s hysterezí.

Zvolíme tedy dva prahy, minimální  $T1$  a maximální  $T2$ . Tyto prahy lze nastavit automaticky dle odhadu poměru signálu od šumu. Je-li hodnota gradientu nižší než  $T1$ , je gradient brán jako nehranový. Je-li hodnota vyšší než  $T2$ , je okamžitě označen jako hrana. Nachází-li se gradient mezi mezními hodnotami  $T1$  a  $T2$ , pak je bod označen jako hranový pouze v případě, je-li jeho sousedem bod, který už byl označen jako hrana dříve.

## 4 NÁVRH ŘEŠENÍ

Většina uvedených zabezpečovacích praktik zcela jistě pomáhá při využívání formátu Motion JPEG 2000, respektive JPEG 2000, z hlediska bezpečnosti. Každý ze způsobů je určen pro určitou množinu vzniklých chybných stavů. Uvedené zabezpečovací metody, implementované přímo do specifikace standardu, jsou zvoleny pro primární způsoby zabezpečení již ve fázi kompresního algoritmu. Naproti tomu uvedené maskovací metody řeší chybové stavy na opačném konci využití tohoto formátu, tedy opravu chyb vzniklých při přenosu na straně příjemce multimediálního obsahu. Na tyto chybové stavy jsou implementované metody nedostatečné.

Ať už jde o techniky z 1. části standardu, kde se mluví o základních metodách (například reinicializaci) nebo posléze uváděné sofistikovanější metody z 8. části (JPSEC), respektive 11. části (JPWL), ve všech případech se jedná o implementované metody probíhající uvnitř kompresního algoritmu. Při řešení zabezpečovacích problémů byly v našem případě použity zbývající zmíněné techniky - maskovací metody. Díky zaměření algoritmů v knihovnách dostupných v [9], byly pro řešení popisovaného problému chyb, vznikajících při přenosu, vybrány některé z technik implementovaných do testovacího programu ECM (Error Concealment Methods), vytvořeného za účelem aplikace maskovacích metod. Jedná se o software vytvořený v programovém prostředí Matlab, které se ukazuje jako nejvhodnější řešení takto koncipovaných řešení díky široké podpoře matematických algoritmů. Umožňuje návrh grafického uživatelského prostředí (GUI), kterým se výsledný testovací program vyznačuje.

Formát Motion JPEG 2000 je sled snímků skládajících se z jednotlivých obrazů komprimovaných metodou JPEG 2000. Jedná se tedy o metodu intra-frame jak bylo uvedeno v Úvodu. Díky této skutečnosti je v prostředí Matlab možno s takovouto videosekvencí pracovat jako s omezenou posloupností obrazů ve formátu JPEG 2000. Matlab tento formát nepodporuje, proto se v testovacím programu ECM využívá externí program určený pro kompresi a dekompresi formátu JPEG 2000. Jedná se o program Kakadu [24]. Umožňuje simulaci chyb. Při simulaci vychází z vlastností bezdrátových sítí Wi-Fi, u kterých je z důvodů rizika ztráty celého datového rámce voleno menších rámců, pro minimalizaci toho problému. Délka je tedy 512B. Je možné simulovat bitové chyby, ztráty paketů, atd.

## 4.1 Zvolené zabezpečovací metody

Pro řešení a následné testování byly použity tyto metody:

- Metoda LL Subband
- Metoda filtrační pomocí filtrů
  - Průměrovací filtr (lineární)
  - Kruhový filtr (lineární)
  - Mediánový filtr (nelineární)
- Metoda Nahrazování
- Metoda obnovení hran pomocí hranových detektorů
  - Cannyho hranový detektor
  - Sobelův detektor
  - Logaritmický detektor
  - Kombinace výše uvedených

U uvedených metod lze určit další parametry. U metody LL Subband lze definovat snímek použitý pro korekci (stejně tak i v metodě Nahrazení) a úroveň DWT. U filtračních metod lze specifikovat velikost okna.

Testování na zvolených obrazech, respektive videosekvencích, je uvedeno v následující kapitole. Jsou zde uvedeny i výsledky měření objektivní kvality a porovnání jednotlivých zvolených řešení v obrazové formě.

## 5 TESTOVÁNÍ NA ZVOLENÝCH VIDEOSEKVENCÍCH

Testování proběhlo pro všechny zvolené metody na vybraných sekvencích. Videosekvence, respektive jednotlivé obrazy pro demonstraci filtračních metod a hranových detektorů, byly voleny v různé skladbě expozice, aby byla názorně předvedena odlišnost daných metod pro jednotlivé techniky a jejich závislost na konkrétním případě expozice a charakteru chyb. Simulovány byly různé druhy chyb. U většiny testování proběhla korekce parametrů charakteristických pro průběh dané maskovací metody, aby se projevil vliv těchto veličin na výsledný rekonstruovaný obraz.

### 5.1 Metoda LL Subband

Testovací videosekvence číslo 1:



Originální snímek videosekvence



Poškozený snímek – výrazné barevné zvlnění,  
PSNR: 10.41



Opravený snímek – korekční snímek  $n-1$ , úroveň DWT 4, PSNR: 17.66

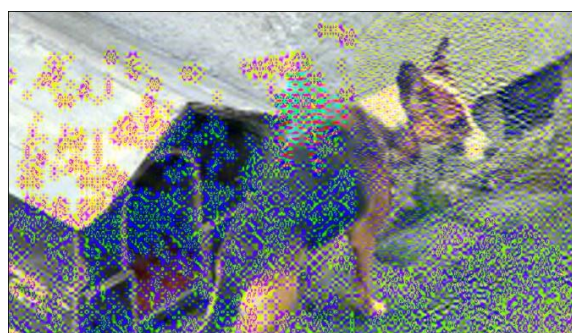


Při uvedeném testování videosekvence č. 1 se projevuje výrazné zlepšení objektivní měřitelné kvality PSNR, která je uvedena u jednotlivých obrázků (poškozený snímek PSNR=10.41, opravený snímek PSNR=17.66). Dochází také k výraznému zlepšení subjektivní kvality jako vjemu pro člověka. Při metodě byl jako korekční snímek použit snímek předchozí, tedy  $n-1$ . Nejlepšího výsledku bylo při testování metody LL subband dosaženo s úrovní DWT 4, kdy si obraz zachová část svých detailů. Výsledek ostatních nastavení lze vidět na přiloženém CD.

### Testovací videosekvence číslo 2:



Originální snímek videosekvence



Poškozený snímek – barevná deformace,  
PSNR: 10.51



Opravený snímek – korekční snímek  $n-1$ , úroveň DWT 2, PSNR: 14.31

Podobně u videosekvence č. 2 dochází při rekonstrukci k požadovanému zvýšení kvality. V tomto případě je uvedena varianta, kdy je kvalita, respektive hodnota PSNR, zvýšena za cenu částečného nahrazení snímkem předchozím v některých detailech. K tomu dochází při úrovních DWT nižších než 4. Jako korekční snímek byl zvolen snímek  $n-1$ . Opět je rozdíl lépe patrný při porovnání na přiloženém CD.

**Testovací videosekvence číslo 3:**

Originální snímek videosekvence

Poškozený snímek – výrazné poškození,  
PSNR: 18.81Opravený snímek – korekční snímek  $n-1$ ,  
úroveň DWT 2, PSNR: 19.59Opravený snímek – korekční snímek  $n-2$ ,  
úroveň DWT 7, PSNR: 18.8

V tomto případě bylo ověřeno, že při zvolení korekčního snímku  $n-2$  nedojde k výraznému zlepšení PSNR, jelikož není využita vysoká podobnost mezi po sobě jdoucími snímky. Proto je vhodnější používat ve většině případů snímek  $n-1$ , obzvláště v dynamicky se měnících videosekvencích, u kterých dochází k výrazné změně expoziční scény mezi dvěma snímky. Díky zvolení dynamické sekvence je názorně ukázán problém, uvedený při předchozím testování, a to použití DWT úrovně nižší než 4 (konkrétně úrovně 2), při kterém dochází k prostému nahrazení snímků. Na poloze ocasu u psa na 3. obrázku je patrné, že za cenu zvýšení kvality dochází ke ztrátě detailů konkrétního snímku (rozdíl oproti originálnímu snímku je nejvýraznější právě v poloze ocasu). Oproti tomu i při použití snímku  $n-2$  na 4. obrázku, tentokrát s DWT 7 úrovně, je dle



polohy ocasu vidět, že snímek si zachoval více ze svých detailů, ale nedošlo k tak výrazné rekonstrukci chyb v obraze. Díky tomu je PSNR nižší.

## 5.2 Metoda filtrační



Originální snímek videosekvence



Poškozený snímek – poškození,  
PSNR: 19.75



Opravený snímek – průměrovací filtr, velikost  
okna 5, PSNR: 25.23



Opravený snímek – průměrovací filtr, velikost  
okna 8, PSNR: 26.02



Opravený snímek – kruhový filtr, velikost okna 2, PSNR: 24.72



Opravený snímek – kruhový filtr, velikost okna 5, PSNR: 24.86



Opravený snímek – mediánový filtr, velikost okna 3, PSNR: 23.16



Opravený snímek – mediánový filtr, velikost okna 5, PSNR: 24.98

Při testování filtračních metod byly postupně aplikovány filtry průměrovací, kruhový a mediánový. Pro názornost byl při jejich užití nastavován odlišný parametr velikost okna. U všech druhů filtrů je proto uvedena dvojice zrekonstruovaných obrazů k porovnání vlivu tohoto parametru. Při zvyšování velikosti okna dochází ve všech parametrech k výraznějšímu rozmazání obrazu v porovnání s menší velikostí okna. Přesto je vždy hodnota PSNR u většího okna vyšší, což je dáno charakterem zvolené chyby viditelné na poškozeném obraze. Nejvyšší nárůst objektivní kvality byl v tomto případě naměřen při použití průměrovacího filtru s oknem o velikosti 8, kde je hodnota PSNR=26.02 (pro srovnání PSNR poškozeného obrazu = 19.75). Aplikace filtru na jinou expozici obrazu a pro jiný typ chyby je uveden v příloze č. 1.



### 5.3 Metoda Nahrazování



Originální snímek videosekvence

Zcela zničený snímek, došlo by k narušení kontinuity videosekvence, dochází proto ke kompletnímu nahrazení snímkem předchozím.



Opravený snímek – korekční snímek  $n-1$

Jak bylo uvedeno v kapitole zabezpečovacích metod, pro udržení spojitosti videosekvence je nutné zajistit náhradu zcela poškozeného snímku, vzniklého například výrazným poškozením nebo ztrátou hlavičky souboru. Díky zvolení stejné dynamické videosekvence jako při testování metody LL subband č. 3 je zde vidět spojitost s tímto testováním v porovnání polohy ocasu psa. Dochází tedy k prostému nahrazení snímků, čemuž máme při metodě LL subband snahu se vyvarovat použitím vyšší úrovně DWT. Kontinuita videa je samozřejmě celistvější, použijeme-li pro nahrazení snímek  $n-1$ .

## 5.4 Metody hranových detektorů



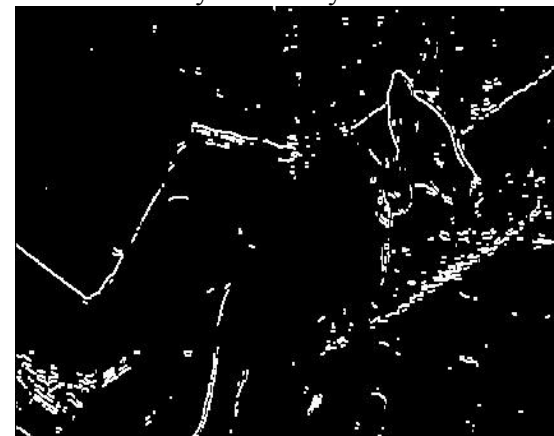
Originální snímek videosekvence



Cannyho hranový detektor



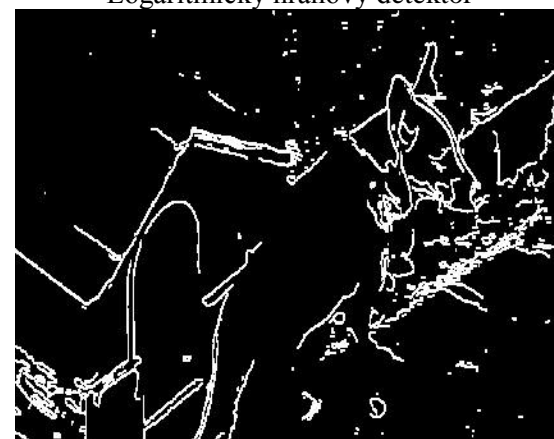
Sobelův hranový detektor



Logaritmický hranový detektor



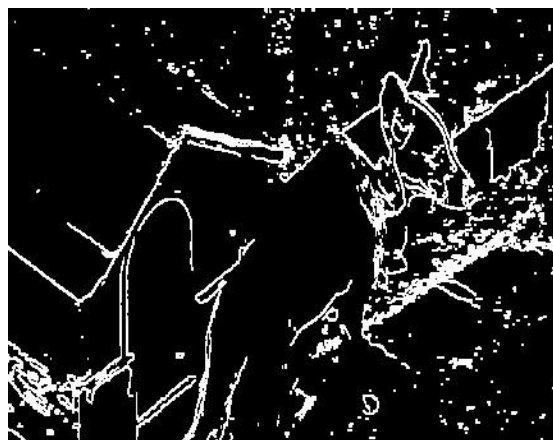
Cannyho + Sobelův detektor



Cannyho + Logaritmický detektor



Sobelův + Logaritmický detektor



Cannyho + Sobelův + Logaritmický detektor

Testování hranových detektorů prokázala značné odlišnosti použitých algoritmů. Porovnáme-li v daném případě výsledky použití Sobelova, Cannyho a Logaritmického detektoru z hlediska subjektivní kvality detekce hran, nejlépe lze zhodnotit Cannyho hranový detektor, z jehož výsledku lze s nejmenší snahou rozeznat obsah původního obrazu díky citlivé detekci při prahování a eliminaci bezvýznamných hran. Další příklady potom ukazují kombinace těchto tří detektorů, kterými lze získat mnohem výraznější podobnosti, tedy lepší detekci hran. Jiný příklad aplikace hranových detektorů pro jinou skladbu objektů a hran je uveden v příloze č. 3.

## 5.5 Další výsledky

Při dalším testování se ukázalo, že uvedené metody je vhodné kombinovat, což zjednodušeně ukazuje kombinace hranových detektorů, při níž se dosahuje lepších výsledků. Podobně je vhodné aplikovat vybraný filtr následně po metodě LL subband. Chyby, charakteristické pro filtraci obrazu, metoda LL subband neodstraní, proto je doplněna vhodným filtrem. Stejně tak se nabízí doplnění filtračních metod o hranové detektory, které částečně potlačí negativní projev filtrů na hranách v obraze. To je ukázáno v příloze č. 2.

Všeobecně pro zlepšení či zhoršení objektivního hodnocení kvality PSNR platí, že vývoj této veličiny vždy nesouhlasí se subjektivním hodnocením, tedy s lidským pocitem a vjemem ze zrekonstruovaného obrazu. Všechny výše uvedené výsledky testování jsou mnohem lépe patrné v jejich elektronické podobě na přiloženém CD. Díky možnosti porovnávat za sebou jdoucí snímky postupně pro všechny úrovně DWT je lépe vidět vliv tohoto parametru na detaily obrazu. Další testy jsou pro zachování celistvosti textu uvedeny v příloze na konci práce.

## 6 NÁROČNOST PRO MOBILNÍ TECHNOLOGIE

Veškeré uvedené testování probíhalo na přenosném počítači, tedy na mobilním zařízení. Z hlediska zhodnocení vhodnosti užití takových matematických operací jako je DWT, jsou v dnešní době rozhodující výkonnostní rozdíly mezi jednotlivými zařízeními. Tím se rozumí strmě narůstající výkon notebooku, dokonce i kapesních počítačů PDA, u kterých již není výjimkou procesor s frekvencí přesahující hranici 1GHz. Dochází díky tomu k velkým výkyvům výkonnosti běžně dostupných zařízení. Proto generalizovat tvrzení, že pro mobilní zařízení je použití uvedených metod vhodné či nevhodné, je nesprávné. Jiným problémem je shodné hodnocení výpočetní náročnosti uvedených algoritmů. Například časovou náročnost výpočtu u ztrátové varianty komprese nelze přesně určit, jelikož zcela závisí na složení původního obrazu a na množství detailů, které obsahuje [25]. Obecně při určování výpočetní náročnosti waveletové transformace platí, že se zvyšujícím se počtem koeficientů se zvyšuje i výpočetní náročnost, tedy doba provedení transformace. Při použití uvedených maskovacích metod dochází v podstatě k nejsložitější operaci při DWT, respektive IDWT, protože například metodu LL subband lze aplikovat prakticky po rozkladech dvou obrazů na subpásma pomocí DWT.

Při návrhu DCT v původní JPEG kompresi byl velký důraz kladen na co nejmenší výpočetní i paměťovou náročnost. Na rozdíl od toho DWT transformace, je výpočetně a paměťově složitým algoritmem. Pro částečné řešení tohoto problému existuje řešení zvané lifting schéma, které ve formě změněného algoritmu průběhu DWT nabízí značnou paměťovou i výpočetní úsporu. Výhoda lifting schématu tkví v principu, při kterém je výpočet aplikován přímo na vstupní koeficienty a výsledky samostatných částí výpočtů nahrazují v paměti tyto původní koeficienty [8]. Tím je dosaženo značného snížení počtu kroků výpočtu a zároveň je minimalizována paměťová náročnost. Například lifting schéma Daubechies 9/7 filtru je díky této úpravě DWT algoritmu o 64% rychlejší než běžná implementace [12]. Snížení výpočetní náročnosti lze dosáhnout i díky užití Haarovy vlnky místo Daubechies vlnky.

Tímto krokem výrazného snížení výpočetní náročnosti klíčové části kompresního řetězce, tedy DWT, dochází ke snížení náročnosti celého JPEG 2000 algoritmu. Tento jev se tedy projeví i při užití maskovacích metod. Při vlastním testování byla ověřena bezproblémová funkčnost metod na mobilním zařízení. Implementace programu ECM navíc není navržena pro nízkou výpočetní náročnost, ale pro efektivitu a názornost při testování maskovacích metod. S těmito poznatky potom lze zjednodušeně tvrdit, že takováto implementace nebude činit notebookům problémy. Pro



respektování slov v úvodu této kapitoly ale musí následovat upřesnění, že tak tomu bude při použití moderních zařízení odpovídajícím dnešním standartům. Zvláště díky trendu strmě rostoucích výkonů u všech typů zařízení, mobilní zařízení nevyjímaje. Ten zajišťuje, že ve většině případů je výkonnost těchto zařízení pro standartního uživatele až předimenzovaná.

## 7 ZÁVĚR

Motion JPEG 2000 je jako kompresní algoritmus založen na intra-frame kódování. Tato metoda znamená, že videosekvence v tomto formátu je sestavena z jednotlivých snímků samostatně komprimovaných ve formátu JPEG 2000. Není využito technik, které využívají ke kompresi závislost a podobnost snímků a díky tomu výrazně snižují množství redundantních dat. Je ovšem možné užít některé nové žádané vlastnosti, jimiž disponuje právě JPEG 2000.

V první části textu jsem podrobně rozebral novinky a výhody tohoto standardu, jeho jednotlivé specifikace a srovnání se starším předchůdcem JPEG. Dále byl detailně popsán průběh všech částí komprese standardu JPEG 2000. Díky použití DWT je dosaženo některých charakteristických vlastností. Jedná se například o podporu progresivního přenosu informace, jak z pohledu prostorového rozlišení, tak i kvality dekomprimovaného obrazu. Jako další odlišnost oproti staršímu JPEG 2000 lze označit podporu ztrátové a bezztrátové varianty komprese, nabízející možnost umístění obou typů současně v jednom komprimovaném datovém toku. Nelze opomenout odstranění známých blokových artefaktů vznikajících při diskrétní kosinově transformaci při JPEG kompresi. Následující kvantování DZQ a popis použitého entropického kódování EBCOT ukončují první část textu věnovanou průběhu komprese JPEG 2000.

Problém samotného zabezpečení sekvence dat byl rozdělen na dvě oblasti. Jedná se v první řadě o techniky implementované v jednotlivých částech standardu, tedy o primární zabezpečení v 1. části Core Coding System, dále o techniku JPSEC v 8. části a nakonec o metodu JPWL v části 11. Všechny tyto metody jsou určeny k primárnímu zabezpečení datového toku, které má předcházet samotnému vzniku chyb při cestě po přenosovém kanálu s výjimkou JPWL. Tento způsob umožňuje progresivně reagovat na chyby i po průchodu chybným přenosovým prostředím díky označení chybové citlivosti a zbytkových chyb. V další části byla popsána pro nás důležitější technika použití maskovacích metod, které byly podrobeny testům. Tyto nemají za úkol data chránit při přenosu, ale jejich cílem je minimalizování následků způsobených těmito chybami. Jejich užití je vhodné, není-li díky nedostatečné šířce přenosového pásma možné užití jiného zabezpečení přenášených dat. Metoda LL subband, která byla uvedena jako první, je pro tuto činnost zpracování videosekvencí nepostradatelná, což prokazují později provedené testy. Použití dalších uvedených metod jako filtrování a detekce hran ji mají za úkol doplňovat, pouze v některých vhodných případech nahrazovat. I přes to jim v textu byla věnovaná pozornost a byly uvedeny jednotlivé metody filtrační: průměrovací, kruhová, mediánová. Následovalo přiblížení

funkce hranových detektorů: Robertsův operátor, Operátor Prewittové, Sobelův operátor, Robinsonův operátor, Kirschův operátor, Laplaceův operátor, Cannyho hranový detektor.

Pro testovací část byly vybrány videosekvence, jak se staticky, tak i s dynamicky se měnící scénou, čímž je lépe dosaženo názornosti uvedených obrazových výsledků. Všechny zvolené metody pro testování byly uvedeny v kapitole 4.1. Náhled testovaných videosekvencí je vždy spojen s výpisem naměřené objektivní kvality PSNR. V testech byl uveden originální obraz bez poškození, dále simulovaný poškozený obraz s charakteristickými chybami a poté již samotné rekonstruované obrazy po aplikaci jednotlivých zvolených maskovacích metod. Při testování metody LL subband byl v některých případech ověřen značný nárůst kvality po její aplikaci. Bylo také zjištěno, že při příliš nízké zvolené úrovni DWT dochází sice ke zlepšení vizuální kvality poškozeného obrazu za cenu ztráty původních detailů a jejich nahrazení předchozím snímkem. To je patrné při testování č. 3. Při vysoké úrovni se potom neprojevuje zlepšení kvality. Také se prokázala vhodnost použití korekčního snímku  $n-1$ .  $n-2$  již vykazuje větší odlišnost od poškozeného obrazu. Toto platí hlavně v případě dynamických videosekvencí. Díky volbě ukázkových sekvencí jsou všechny uvedené poznatky dobře patrné i na malých obrazových náhledech uvedených v kapitole testování. Při použití filtračních metod se ukázalo vhodné spojení s hranovými detektory, které dokážou obnovit hrany potlačené filtry. Toto spojení je v budoucích aplikacích jistě nutné. Bylo zjištěno, že maskovací metody vždy nezaručí výrazné zlepšení kvality, ale je-li to umožněno ovlivňujícími faktory, potlačí chyby vzniklé přenosem. Těmito faktory můžeme rozumět volbu aplikované metody, nastavení metody, charakter chyb nebo intenzitu chyb. Testováním jsme se seznámili s chybovostí formátu Motion JPEG 2000 a také s problémem zabezpečení, který je v oblasti IT palčivým problémem. Je tedy nutné jej zajistit i v případě komprese Motion JPEG 2000. Přínosem této práce jsou uvedená zjištění a nabyté zkušenosti.

V poslední části textu byla vyslovena úvaha pro náročnost uvedených metod s ohledem na použití v mobilních zařízeních. Pro řešení této části práce a otestování této problematiky v praxi bylo veškeré vlastní testování s programem ECM v Matlabu prováděno právě na mobilním přenosném počítači, na kterém nebyl zaznamenán sebemenší výkonnostní problém. Přesto uvádím možnosti jak výpočetní náročnost v budoucích implementacích snížit užitím tzv. lifting schématu pro DWT transformaci. Právě tuto substituci za běžnou DTW je vhodné do praktických implementací v budoucích aplikacích přidat.

## SEZNAM POUŽITÉ LITERATURY

- [1] JPEG. About our committee. *JPEG Committee*. [Online] 2009.  
<http://www.jpeg.org/committee.html>.
- [2] ITU-T Recommendation, T-800 and ISO/IEC 15444-1. *JPEG2000 Image Coding System: Core Coding System (JPEG2000 Part 1)*. 2000.
- [3] JPEG2000. Our New Standard! *JPEG 2000*. [Online] 2009.  
<http://www.jpeg.org/jpeg2000>.
- [4] Kváš, M. *Kompresa JPEG 2000 a akcelerace pomocí DSP*. [PDF Dokument] 2008.
- [5] JPEG. *JPEG 2000: Part 1 Final Committed Draft*. [PDF Dokument] 2000.
- [6] Levický, Dušan. *Multimediálne telekomunikácie*. Košice : Elfa, s.r.o., 2002. ISBN 80-89066-58-5.
- [7] Taubman D. S., Marellin M. W. *JPEG2000 Image compression, fundamentals, standards and practice*. Massachusetts : Kluwer Academic Publisher, 2004. ISBN 0-7923-7519-X.
- [8] Matela, J. *Implementace JPEG2000 komprese na GPU, Diplomová práce*. [PDF Dokument] 2009.
- [9] Kyselý, F. *Metody maskování chyb pro Motion JPEG2000*. [PDF Dokument] 2008.
- [10] Kozumplík, Jiří. *Vlnkové transformace a jejich využití pro filtraci signálů EKG, Habilitační práce*. Brno : VUTUM, 2004. ISSN 1213-418X.
- [11] Giesl, J. *Fraktální komprese statických obrazů pomocí waveletovy transformace*. [PDF Dokument] 2007.
- [12] Daubechies, I., Sweldens, W. *Factoring Wavelet Transforms into Lifting Steps*. [PDF dokument] 1997.
- [13] Marcellin, W. M., Bilgin, A., Gormish, M., Boliek, M. P. *An Overview of JPEG-2000*. [PDF Dokument] 2000.
- [14] ITU-T Recommendation, T-802 and ISO/IEC 15444-3. *JPEG2000 Image Coding : Motion JPEG2000 (JPEG2000 Part 3)*. 2005.

- [15] JPEG. *JPEG 2000: Part 3 Final Committed Draft*. [PDF Dokument] 2001.
- [16] Bodeček, K., Daněček, P., Vrba, K. *Bezpečný JPEG2000*. [PDF Dokument] 2006.
- [17] Prchal, J., Šimák, B. *Digitální zpracování signálů v telekomunikacích*. Praha : Vydavatelství ČVUT, 2001. ISBN 80-01-02149-1.
- [18] Hlaváč, V. *Komprese Obrazů*. [PDF Dokument] 2009.
- [19] Valenta, T. Odolnost formátu JPEG 2000 proti přenosovým chybám. *Access server*. [Online] Březen 2008. <http://access.feld.cvut.cz/view.php?cisloclanku=2008030002>.
- [20] Vašíček, Z. *Obvodová realizace vyvíjejících se systémů, Diplomová práce*. [PDF Dokument] 2006.
- [21] Linka, A., Volf, P., Košek, M. Zpracování obrazu a jeho statistická analýza. *E-learning*. [Online] Květen 2004. <http://e-learning.tul.cz/cgi-bin/elearning/elearning.fcgi>.
- [22] Hlaváč, V. *Hledání Hran*. [PDF Dokument] 2009.
- [23] Baránek, J. *Metody detekce a reprezentace hran v obraze, Bakalářské práce*. [PDF Dokument] 2007.
- [24] Kakadu. Documentation. *Kakadu Software*. [Online] <http://kakadusoftware.com/>.
- [25] Malý, J. *Srovnání metod pro ztrátovou kompresi obrazu*. [PDF Dokument] 2006.
- [26] Hýna, P. *Detekce rohů v obraze, Bakalářské práce*. [PDF Dokument] 2007.
- [27] Zargari, F., Hashemi, R. M. *An Improved Motion JPEG2000 Decoder for Error Concealment of Segmentation Symbol Faults*. [PDF Dokument] 2008.
- [28] Bodeček, K. Komprese rozměrných obrazů pomocí JPEG2000. *Elektrorevue.cz*. [Online] 2005. <http://www.elektrorevue.cz/clanky/05011/index.html>.

## SEZNAM OBRÁZKŮ

Obr. 1: Ilustrativní srovnání kompresí JPEG a JPEG 2000 .....	6
Obr. 2: Blokové schéma kompresního algoritmu JPEG 2000 [6] .....	7
Obr. 3: Rozložení snímku na dlaždice [6] .....	8
Obr. 4: Jednoduchá jednorozměrná 1D DWT analýza .....	11
Obr. 5: a) Haarova vlnka, b) Daubechies vlnka [11] .....	11
Obr. 6: Analýza a syntéza 1D DWT [12] .....	12
Obr. 7: Tříúrovňová DWT2 transformace .....	14
Obr. 8: Diskrétní vlnková transformace druhé úrovně (DWT2) [8] .....	15
Obr. 9: DWT2 - zpracování po řádcích a po sloupcích .....	16
Obr. 10: Struktura datového toku [6] .....	18
Obr. 11: Syntaxe segmentové značky SEC [16] .....	24
Obr. 12: Syntaxe segmentové značky INSEC [16] .....	24
Obr. 13: Schéma JPWL systému [3] .....	25
Obr. 14: Princip metody LL subband [9] .....	27
Obr. 15: Ukázka aplikace metody LL subband [9] .....	27
Obr. 16: LSI systém .....	28
Obr. 17: Určení odezvy $h[i,j]$ s konvoluční maskou 3x3 se středem konvoluční masky odpovídající hodnotě E [20] .....	29
Obr. 18: Průměrovací filtr se svou konkrétní konvoluční maskou 3x3 [20] .....	30
Obr. 19: Mediánový filtr se svou konkrétní konvoluční maskou 3x3 [20] .....	32
Obr. 20: Náhled výsledku průměrovacího filtru a mediánového filtru [20] .....	32
Obr. 21: Ilustrace detekce hran .....	33
Obr. 22: Jasové profily [22] .....	35
Obr. 23: Hrany v severním směru – originální obraz, severní hrany, prahované hrany [22] .....	37
Obr. 24: Hrany ve východním směru – východní hrany, prahované hrany [22] .....	37
Obr. 25: Sobelův detektor - originální obrázek, kompletní filtrace, vertikální filtrace, horizontální filtrace .....	38
Obr. 26: Původní obrázek, filtrace operátorem $h_1$ , filtrace operátorem $h_2$ , filtrace operátorem $h_3$ , filtrace operátorem $h_4$ .....	40
Obr. 27: Dekódovatelnost hran dle uvedených kritérií [22] .....	41

## SEZNAM PŘÍLOH

- Příloha č. 1: Výsledek testování – Srovnání filtračních metod
- Příloha č. 2: Výsledek testování – Aplikace hranových detektorů následně po filtraci
- Příloha č. 3: Výsledek testování – Hranové detektory

## PŘÍLOHA Č. 1



Originální snímek videosekvence



Poškozený snímek – PSNR: 21.58



Opravený snímek – Průměrovací filtr, velikost okna 4, PSNR: 23.4



Opravený snímek – Průměrovací filtr, velikost okna 8, PSNR: 25.76



Opravený snímek – Kruhový filtr, velikost okna 2, PSNR: 23.37



Opravený snímek – Kruhový filtr, velikost okna 5, PSNR: 26.07



Opravený snímek – mediánový filtr, velikost okna 5, PSNR: 23.69



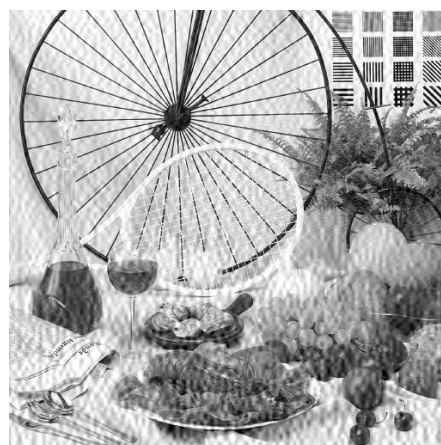
Opravený snímek – mediánový filtr, velikost okna 7, PSNR: 25.2



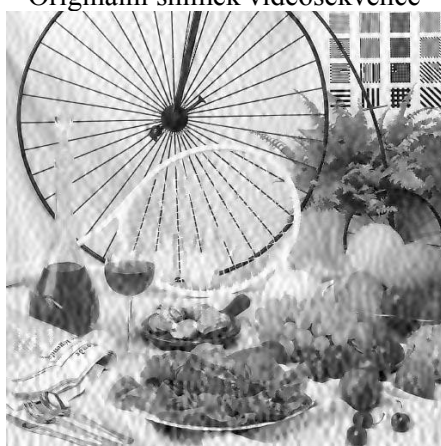
## PŘÍLOHA Č. 2



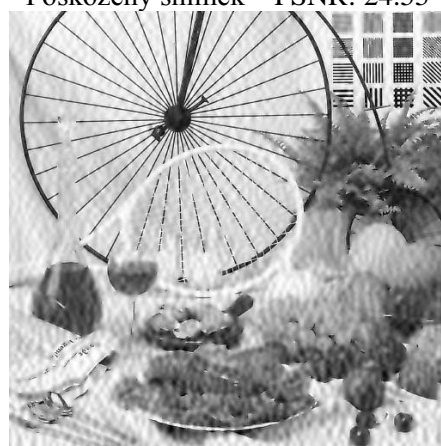
Originální snímek videosekvence



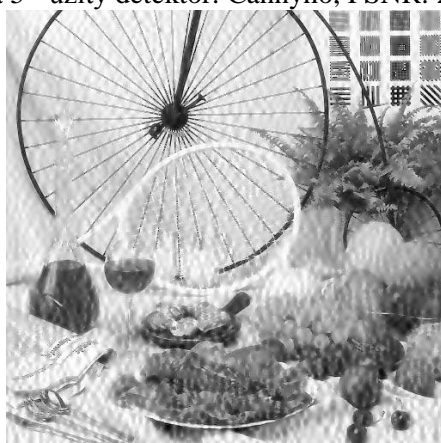
Poškozený snímek – PSNR: 24.55



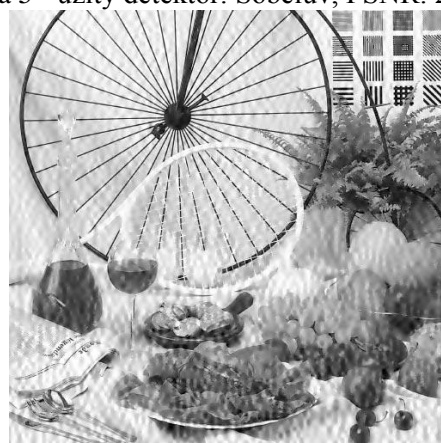
Opravený snímek – Průměrovací filtr, velikost okna 5 - užitý detektor: Cannyho, PSNR: 24.85



Opravený snímek – Průměrovací filtr, velikost okna 5 - užitý detektor: Sobelův, PSNR: 25.06



Opravený snímek – Průměrovací filtr, velikost okna 5 – užitý detektor: Logaritmický, PSNR: 24.79



Opravený snímek – Průměrovací filtr, velikost okna 5 - užitý detektor: Cannyho, Sobelův, Logaritmický, PSNR: 25.49

## PŘÍLOHA Č. 3



Originální snímek videosekvence



Cannyho hranový detektor



Sobelův hranový detektor



Logaritmický hranový detektor



Cannyho + Sobelův detektor



Cannyho + Logaritmický detektor



Sobelův + Logaritmický detektor



Cannyho + Sobelův + Logaritmický detektor